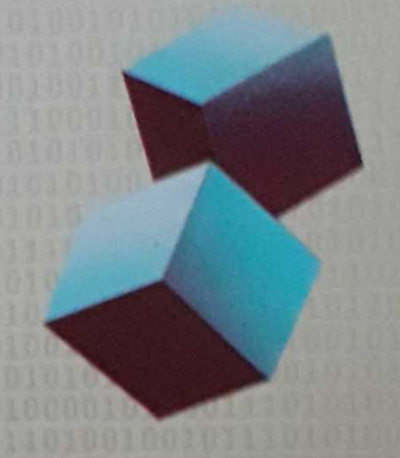
مقدمة في
تنظيم ومعمارية الحاسب الآلي

د. محمد عبدالسلام عزاقة



الطبعة الثانية
2018

مكتبة طرابلس العلمية العالمية

فهرس المحتويات

تمهيد

الفصل الأول

مقدمـــــة

1 - مقـدمـة

يتناول هذا الكتاب بنية و وظائف أجهزة الحاسب الآلى لغرض تقديم شرح عن طبيعة وخصائص أنظمة الحاسب الآلى في العصر الحديث ، وتشكل هذه المهمة تحدياً صعباً لسببين هما :

أولا : هناك مجموعة متنوعة وهائلة من أنظمة الحواسيب التى تبدأ من الحواسيب ذات الشريحة الواحدة بتكلفة بضعة دولارات إلى أجهزة الحاسب العملاقة بتكلفة عشرات الملايين من الدولارات ، فكلها يمكن أن تدعى حق المطالبة بأن تسمى نظام حاسب آلى ، فالتنوع ليس في التكاليف فقط و أنما في الحجم ، والأداء ، والتطبيق.

ثانياً : إن وتيرة التغيير السريع الذي اتسمت به – دائما - تقنية الحاسب الآلى استمرت بلا هوادة ، وغطت جميع جوانبه التقنية من أسس تقنية الدوائر المتكاملة المستخدمة في بناء مكوناته إلى الأستخدام المتزايد لمفاهيم التنظيم المتوازي في الجمع مابين تلك المكونات.

وبالرغم من تنوع وتيرة التغيير في مجال الحاسب ، فإن هناك مفاهيم أساسية معينة تُطبق باستمرار فى نظم الحاسب ، وإن تطبيق هذه المفاهيم يعتمد على الحالة الراهنة للتقنية والسعر/الأداء المستهدف من قبل المصمم ، فالقصد من هذا الكتاب هو تقديم مناقشة عن الأساسيات فى تنظيم الحاسب وهندسته المعمارية ، وربط ذلك بقضايا تصميمه .

يقدم هذا الفصل المنهج المستخدم لوصف وفهم نظام الحاسب الآلى .

1.1 تنظيم ومعمارية الحاسب الآلى

عند وصف أجهزة الحاسب يتم في كثير من الأحيان التمييز بين المعمارية والتنظيم ، فمعمارية الحاسب تشير إلى سمات النظام المرئية للمبرمج أو تلك الصفات التي لها تأثير مباشر على التنفيذ المنطقي للبرنامج ، وتنظيم الحاسب يشير إلى الوحدات التنفيذية وترابطها لكى تحقق المواصفات المعمارية. والأمثلة على السمات المعمارية تشمل فئة أو طقم التعليمات وعدد الخانات المستخدمة لتمثيل مختلف أنواع البيانات (على سبيل المثال : الأرقام و الأحرف) وآليات الأدخال و الإخراج والتقنيات اللازمة لمعالجة الذاكرة. أما السمات التنظيمية، فإنها تشمل تلك التفاصيل المادية (الإلكترونية) الغير مرئية للمبرمج مثل إشارات التحكم و الربط بين الحاسب والأجهزة الطرفية وتقنيات الذاكرة المستخدمة ، فعلى سبيل المثال ، إن التساؤل حول ما إذا كان للحاسب تعليمات للضرب الحسابى أم لا فإنه أمر يختص به التصميم المعماري ، أما من الناحية التنظيمية فيمكن تنفيذ هذه التعليمات من قبل وحدة خاصة بالضرب أو من خلال الية الأستخدام المتكرر لوحدة الاضافة (الجمع) فى النظام . ويمكن أن يستند هذا القرار التنظيمي على التكرار المتوقع لأستخدام تعليمات الضرب والسرعة النسبية للأسلوبين و التكاليف و الحجم المادي للوحدة الخاصة بالضرب .

إن الشركات المصنعة للحاسب تقدم عائلة مكونة من العديد من نماذج الحاسب التى لها نفس المعمارية ولكن مع وجود اختلافات في التنظيم ، وبالتالي فإن نماذج مختلفة في العائلة لها سعر وخصائص أداء مختلفة ، وعلاوة على ذلك فإن معمارية خاصة قد تمتد لسنوات عديدة ، وتشمل عددا مختلفا من النماذج الحاسوبية التى يتغير تنظيمها مع تغير التقنية . ومن الأمثلة البارزة على ذلك

معمارية الحاسب (IBM System/370) حيث قُدم هذا النظام الحاسوبى للمرة الأولى في عام 1970 وشمل عدداً من النماذج بحيث يمكن للزبون ذو المتطلبات المتواضعة شراء نموذج رخيص وبطيء . أما إذا زادت متطلباته فإنه يمكنه لاحقا رفع مستوى الحاسب بآخر أسرع وأكثر تكلفة من دون الحاجة إلى التخلي عن البرامج التي سبق أن كُتبت و صُممت ، فهذه النماذج الجديدة أبقت على نفس المعمارية مما أدى ألى حماية الزبون من الاستثمار فى برمجيات جديدة . إن التغيرات في التقنية لا تؤثر فقط على التنظيم ولكنها تقدم أيضا معمارية أكثر قوة وأكثر تعقيداً ، وعموما فإن الاصرار فى الاجهزة الحاسوبية الصغيرة على التوافق مابين جيل و جيل أمر غير مطلوب.

1.2 الوظيفة والبنية

إن الطبيعة الهرمية لأنظمة الحاسب أسلوب أساسي فى تصميمها وكذلك وصفها ؛ فالمُصمِم يحتاج فقط للتعامل مع مستوى معين من النظام في وقت ما ، وعند كل مستوى يتكون النظام من مجموعة مكونات مترابطة مع بعضها ، ويعتمد عملها على خصائص مبسطة ومحددة للنظام في المستوى السفلي الذى يسبقه .

فعند كل مستوى يهتم المصمم بالبنية والوظيفة :

- البنية : هى الطريقة التي ترتبط بها المكونات .
- الوظيفة : هى عمل أو شغل كل مكون من المكونات كجزء من الهيكل البنائي الكامل للنظام .

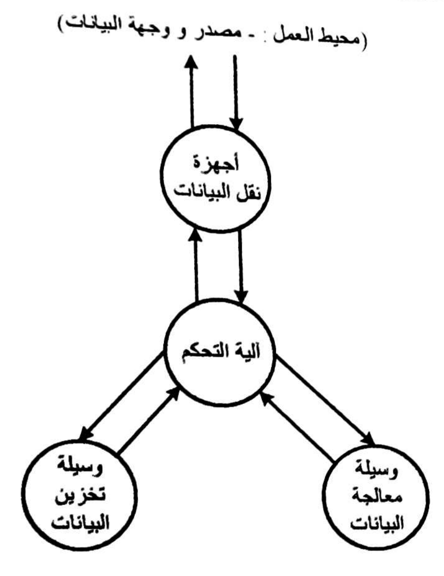
وظيفة نظام الحاسب :

إن وظيفة الحاسب وهيكله البنائى بسيط فى جوهره ، فالشكل (1.1) يصور الوظائف الأربع الأساسية التي يمكن أن يؤديها الحاسب بصورة عامة ، وهى :

- معالجة البيانات .
- تخزين البيانات .
- حركة البيانات .
- التحكم .

فمن الطبيعى أن يكون للحاسب القدرة على معالجة البيانات ، وقد تتخذ هذه البيانات مجموعة واسعة من الأشكال ، وبالتالى فإن متطلبات المعالجة تتسع . ومن الضروري أيضا أن تتوفر للحاسب القدرة على تخزين البيانات حتى وإن كان الحاسب يعالج البيانات بشكل سريع (أي بيانات تدخل ويتم معالجتها و تخرج النتائج فورياً) ، فالحاسب يجب أن يخزّن - مؤقتا على الأقل - تلك القطع من البيانات التي جرى العمل عليها في أي لحظة وبالتالي فإن هناك على الاقل وظيفة لتخزين البيانات للمدى القصير .

وبنفس القدر من الأهمية فإن للحاسب وظيفة لتخزين البيانات على المدى الطويل بحيث يتم تخزين ملفات البيانات على الحاسب لاسترجاعها لاحقا وتحديثها . كذلك يجب على الحاسب أن يكون قادراً على نقل البيانات بينه وبين المحيط الخارجي المتصل به .

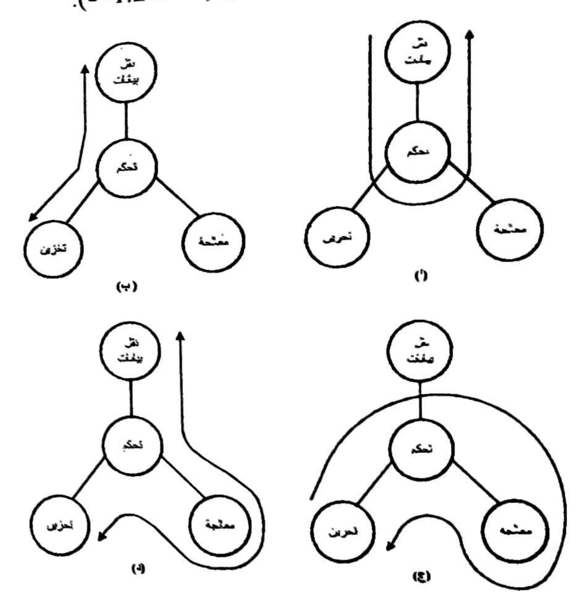


الشكل (1.1) – الشكل الوظيفى للحاسب الآلى

تحتوى بيئة تشغيل الحاسب على أجهزة تعمل إما كمصادر أو وجهات للبيانات ، وتعرف عملية الإدخال/الإخراج بعملية استلام بيانات أو تسليمها إلى جهاز مرتبط مباشرة بالحاسب ، ويشار إلى الجهاز على أنه طرفية . أما عملية نقل البيانات عبر مسافات أطول من / إلى جهاز بعيد ، فتعرّف بأتصالات البيانات.

أخيراً ، يجب أن يكون هناك سيطرة على هذه الوظائف الثلاث . وتدير وحدة التحكم داخل الحاسب موارده المادية ، وتنظم أداء أجزائه الوظيفية استجابة وتنفيذاً للتعليمات. يصور الشكل (1.2) أربعة أنواع ممكنة للعمليات حيث يمكن

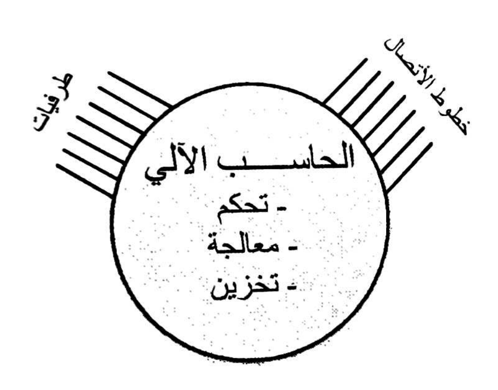
للحاسب أن يعمل كجهاز لحركة البيانات (الشكل 1.2- أ) وذلك لمجرد نقل البيانات من جهاز طرفي أو خط أتصالات إلى آخر. ويمكن للحاسب أن يعمل أيضا كجهاز لتخزين البيانات (الشكل 1.2- ب) من خلال نقل البيانات من المحيط الخارجي الى وحدة تخزين الحاسب (القراءة) والعكس بالعكس (الكتابة). وتبين المخططات النهائية لوظائف الحاسب العمليات التي تنطوي على معالجة البيانات سواء كانت هذه البيانات في وحدة تخزين الحاسب - الشكل (1.2 - ج) - أو في المسار مابين وحدة التخزين والمحيط الخارجي (الشكل 1.2- د).



الشكل (1.2) - العمليات المحتملة لنظام الحاسب الالى

بنية نظام الحاسب:

يبين الشكل (1.3) أبسط تصور ممكن لجهاز الحاسب الذى يتفاعل بطريقة ما مع محيطه الخارجي ، و بشكل عام يمكن تصنيف جميع صلاته مع المحيط الخارجي إما كأجهزة طرفية أو خطوط اتصالات.

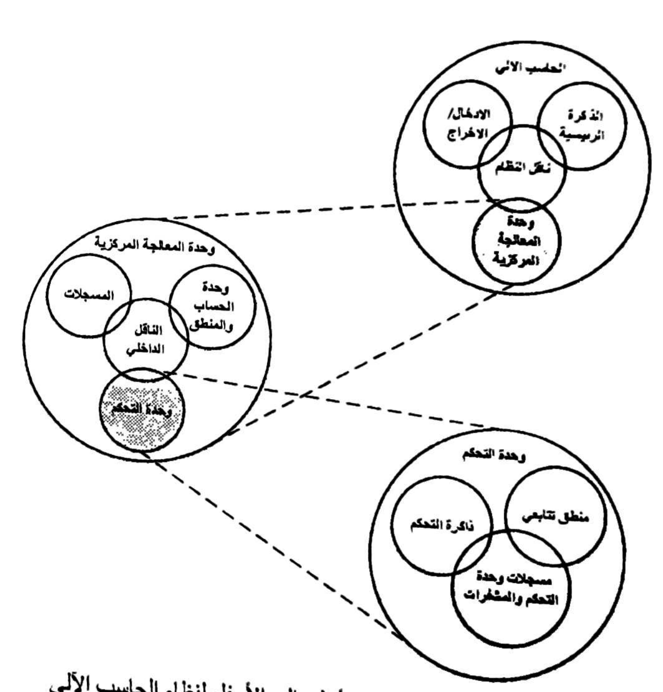


الشكل (1.3) – نظام الحاسب الآلى

في البنية الداخلية للحاسب - والتى تظهر في الشكل (1.4) - **هناك أربعة عناصر** هيكلية رئيسة :

- وحدة المعالجة المركزية : **تتحكم في تشغيل جهاز الحاسب ، وتؤدي وظائفه في معالجة البيانات ، و يشار إليها بالمعالج.**
- الذاكرة الرئيسية : **تقوم بتخزين البيانات .**
- الإدخال/الإخراج : **تقوم بنقل البيانات بين الحاسب ومحيطه الخارجي .**

- نظام الربط البينى : هو الآلية التي تمكّن من الاتصال بين وحدة المعالجة المركزية و الذاكرة الرئيسة والإدخال/الإخراج . ومن الأمثلة الشائعة لربط مكونات النظام هي إستخدام ناقل النظام و الذى يتألف من عدة أسلاك توصيل تربط مكونات الحاسب ببعضها .



الشكل (1.4) - مسقط من الأعلى الى الأسفل لنظام الحاسب الآلى

قد يكون هناك عنصراً واحد أو أكثر من العناصر المذكورة أنفاً بالحاسب ، وتقليدياً يوجد فى الحاسب معالج واحد فقط . وفي السنوات الأخيرة هنالك استخدام متزايد لمعالجات متعددة في جهاز حاسب واحد.

أما العنصر الأكثر إثارة للاهتمام والأكثر تعقيداً في الحاسب هو وحدة المعالجة المركزية ، ومكوناتها الهيكلية الرئيسية هي كما يلي:

- وحدة التحكم : التحكم في تشغيل وحدة المعالجة المركزية ، وبالتالي الحاسب .
- وحدة الحساب والمنطق : تؤدي وظائف الحاسب فى معالجة البيانات .
- المسجلات : توفر سعة تخزين داخلية لوحدة المعالجة المركزية .
- رابط وحدة المعالجة المركزية : الآلية التي توفر الاتصال بين وحدة التحكم و وحدة الحساب والمنطق والمسجلات .

1.3 التسلسل الهرمي لمستويات نظام الحاسب

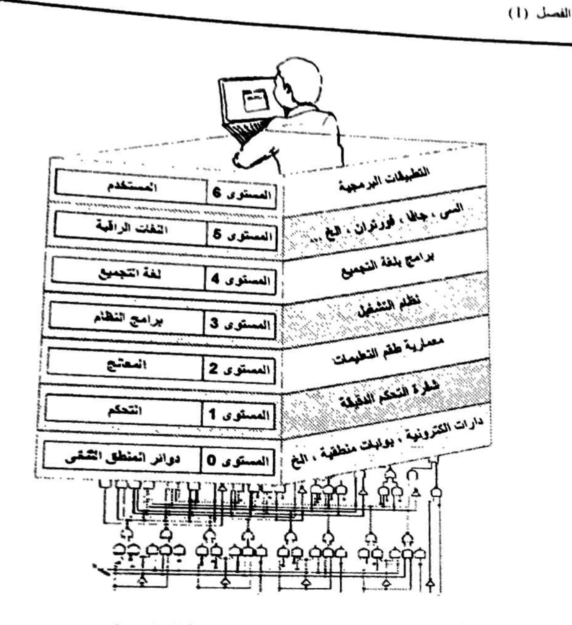
حتى يتمكن نظام الحاسب من حل مجموعة واسعة ومتنوعة من المعضلات يجب أن تكون له القدرة على تنفيذ البرامج المكتوبة بلغات برمجة مختلفة مثل (FORTRAN) و (C) و (LISP) و (PROLOG) وغيرها من لغات البرمجة الراقية . ونظام الحاسب متكون مادياً (إلكترونياً) من بوابات ودوائر منطقية ، واشارت كهربية وأسلاك ربط ، وذلك سبب فى وجود فجوة هائلة مابين هذه المكونات المادية ولغات راقية المستوى مثل (C++) ، وحتى يكون النظام عملياً يجب أن تكون هذه الفجوة غير مرئية لمعظم مستخدمي النظام .

من المبادئ الأساسية في البرمجة أنه عندما تكون المشكلة كبيرة يجب تجزئتها الى أجزاء مستخدمين فى ذلك منهج "فرق تسد" . ففي البرمجة يتم تقسيم المشكلة إلى أجزاء او مُركبات ويتم تصميم كل مُركب على حدة ، وكل مُركب يقوم بأداء مهمة معينة ويحتاج المُركب فقط الى معرفة كيفية الارتباط مع المُركبات الأخرى لإستخدامها أو التفاعل معها .

ويمكن تناول تنظيم نظام الحاسب بطريقة مماثلة وذلك من خلال مبدأ هيكلية المستويات الهرمية بحيث يمكننا أن نتصور أن النظام عبارة عن هرم من المستويات وكل مستوى لديه وظيفة محددة ويعتبر لوحده كآلة افتراضية ، ويمكن ان نسمي الحاسب الافتراضي في كل مستوى آلة أفتراضية . الآلة الإفتراضية في كل مستوي تنفذ مجموعة معينة من التعليمات الخاصة بها ، وتستدعي الآلات التى في المستويات الأدنى منها للقيام بمهام عند الضرورة . ومن خلال دراسة تنظيم الحاسب يمكن تبيان الأساس المنطقي وراء التقسيم الهرمي لنظام الحاسب، وكذلك كيفية إنجاز وتصميم طبقات هذا التقسيم والتفاعل فيما بينها ، ويبين الشكل (1.5) الطبقات التى تمثل هيكلية الآلات الإفتراضية .

المستوى 6 وهو مستوى المستخدم ، ويتكون من التطبيقات البرمجية وهو المستوى المعروف من الجميع . وفى هذا المستوى يتم تشغيل البرامج التطبيقية مثل معالجات النصوص ، وبرمجيات الرسومات أو الألعاب ، والمستويات الدنيا غير مرئية تقريبا من هذا المستوى .

المستوى 5 هو مستوى لغات البرمجة الراقية مثل C ، C++ ، FORTRAN ، JAVA ، PASCAL ، PROLOG ، وهذه اللغات يجب أن تترجم (سواء باستخدام مترجم أو مُفسر) إلى لغة يمكن أن تفهمها الآلة (المعالج) .



الشكل (1.5) - المستويات الهرمية لنظام الحاسب

اللغات المترجمة يتم ترجمتها إلى لغة التجميع ومن ثم الى لغة الآلة (يتم ترجمتها إلى المستوى الأدنى التالي) . فالمستخدم في هذا المستوى يرى القليل جداً من المستويات الدنيا ، ويجب على المبرمج فى هذا المستوى أن يكون على معرفة بأنواع تراكيب البيانات والتعليمات المتاحة لكل نوع من تراكيب البيانات ، ولكن لا يحتاج الى معرفة كيفية إنجاز تلك الأنواع من التراكيب .

مستوى 4 وهو مستوى لغة التجميع ويتضمن نوعاً من أنواع لغة التجميع . وكما ذُكر سابقاً ، فإن اللغات الراقية المستوى تترجم أولاً الى لغة التجميع والتي تترجم

بعد ذلك مباشرة إلى لغة الآلة ، وهذه الترجمة هى واحد الى واحد بمعنى تعليمة بلغة التجميع تترجم إلى تعليمة واحدة بلغة الآلة . ومن خلال وجود مستويات منفصلة نحن نقلص من الفجوة مابين اللغة الراقية المستوى مثل ++C ولغة الآلة الفعلية (التي تتكون من 0 و1) .

المستوى 3 هو مستوى برامج النظام ويتعامل مع تعليمات نظام التشغيل. وهذا المستوى هو المسؤول عن البرمجة المتعددة وحماية الذاكرة وعمليات المزامنة والمهام الهامة المختلفة الأخرى ، وفي كثير من الأحيان يتم تمرير التعليمات المترجمة من لغة التجميع إلى لغة الآلة من خلال هذا المستوى بدون تعديل .

المستوى 2 وهو مستوى معمارية طقم التعليمات (ISA) أو مستوى الآلة (المعالج) ، ويتكون من لغة الآلة وذلك حسب معمارية المعالج المصمم بها نظام الحاسب. البرامج المكتوبة بلغة الآلة الحقيقية للحاسب يمكن تشغيلها مباشرة على نظام الحاسب من قبل الدوائر المنطقية دون أي ترجمة أو تفسير أو تجميع ، وسيتم دراسة معمارية طقم التعليمات لاحقاً .

المستوى 1 وهو مستوى التحكم حيث تقوم وحدة التحكم بفك شفرة التعليمات وتنفيذها بشكل صحيح ، ويتم نقل البيانات الي أين ينبغي أن تكون ومتى طلب ذلك ، فوحدة التحكم تفسر تعليمات المعالج التي تم تمريرها إليها (كل مرة) من المستوى الأعلى مسببة في أن الفعل المطلوب يتم اجراءه .

ويمكن تصميم وحدة التحكم من طريقتين : البرمجة المادية أو البرمجة الدقيقة ، فإشارات التحكم في وحدة التحكم المبرمجة مادياً تخرج من مُكونات مادية (المنطق الرقمي الألكترونى) ، وهذه الإشارات توجه حركة كل من البيانات والتعليمات إلى الوجهات المختصة في النظام . نموذجياً وحدة التحكم المبرمجة

مادياً سريعة جداً لأنها في الواقع من المكونات المادية (دوائر ومُكونات منطقية)، ولكن عند تصنيعها من الصعب جداً تعديلها وذلك لأنها من المكونات المادية .

الخيار الأخر للتحكم هو انجاز التعليمات باستخدام البرمجة الدقيقة ، فالبرمجة الدقيقة هي برامج مكتوبة بلغة منخفضة المستوى يتم تنفيذها مباشرة من قبل الكيان المادى حيث يتم تغذية تعليمات الآلة المنتجة من المستوى 2 لهذا البرنامج الدقيق ، والذى بدوره يفسر التعليمات ويُفعل منطق مادى خاص ينفذ هذه التعليمات ، وغالبا ما تترجم تعليمة واحدة من هذا المستوى الى عدة تعليمات دقيقة (ترجمة تعليمة الآلة الى برنامج دقيق) ، وهنا لاتوجد علاقة واحد الى واحد مابين لغة التجميع ولغة الآلة . إن التصميم بالبرمجة الدقيقة شائع لأنه نسبياً يمكن تعديلها بسهولة ، ومن سيئات البرمجة الدقيقة أنها أضافت مستوى من الترجمة مما ينتج عنه عادةً تباطؤ في تنفيذ التعليمات .

المستوى 0 هو مستوى مكونات المنطق الرقمي حيث توجد المكونات المادية لنظام الحاسب : البوابات المنطقية والدوائر الألكترونية والأسلاك ، وهذه هي اللبنات الأساسية لنظام الحاسب ، وهى مبنية طبقاً لحساب المنطق الثنائى وهو مشترك مابين جميع أنظمة الحاسب .

مصطلحات مهمة

Arithmetic And Logic Unit (ALU)	وحدة الحساب والمنطق
Central Processing Unit (CPU)	وحدة المعالجة المركزية
Computer Architecture	معمارية الحاسب
Computer Organization	تنظيم الحاسب
Control Unit	وحدة التحكم
Input-Output (I/O)	الإدخال/الإخراج
Main Memory	الذاكرة الرئيسية
Processor	المعالج
Registers	المسجلات
System Bus	ناقل النظام
Sequencing Logic	منطق تتابعي
Control Memory	ذاكرة التحكم
Communication Lines	خطوط الأتصال
Peripherals	طرفيات/ملحقات
Processing	معالجة
Storing	تخزين
Control	تحكم
Data Transfer	نقل البيانات
Instruction Set Architecture (ISA)	معمارية طقم تعليمات المعالج
Hardwired	البرمجة المادية
Microprogrammed	البرمجة الدقيقة
Virtual machine	الآلة الأفتراضية
Assembly Language	لغة التجميع
Compiler	المترجم
Interpreter	المفسر
System Software	برمجيات النظام
Digital Logic	المنطق الرقمي

أسئلة للمراجعة

1. بصورة عامة ، ماهو التمييز بين تنظيم الحاسب ومعمارية الحاسب (ناقش) ؟
2. مالقصود ببنية نظام الحاسب و وظيفة نظام الحاسب ؟
3. ماهي الوظائف الرئيسية الأربعة للحاسب والشكل الوظيفي للنظام في كل وظيفة ؟
4. اذكر العناصر الأساسية لبنية الحاسب .
5. بقائمة وجيزة ، حدد العناصر الأساسية لبنية المعالج .
6. اذكر امثلة للتطبيقات البرمجية الشائعة مع توضيح الفئة المستهدفة لكل تطبيق ؟
7. عدد عشر لغات برمجة راقية في مجالات مختلفة ؟
8. ماهى برامج التشغيل الشائعة لأنظمة الحاسوب المكتبى و الهواتف الذكية؟
9. التقنيات الحديثة في تصنيع الألكترونيات الدقيقة وفرت أمكانية تصنيع لمعالجات متنوعة حسب نوع وصنف أنظمة الحاسب ، بالأطلاع على شبكة المعلومات (الأنترنت) عدد أصناف أنظمة الحاسب وماهى المعالجات المصممة له ؟
10. ماهى الدوائر المنطقية الأساسية المستخدمة في تصميم الأنظمة الرقمية؟

الفصل الثاني

تطور الحاسب الآلي والأداء

2 - تطور الحاسب الالي والأداء

بدءاً يتناول هذا الفصل تطور الحاسب الالى من فكرته و وصولا إلى تصنيع اول نموذج له ، والتطور الذى واكبه حتى وصل الى الشكل المعاصرة له .

نبدأ دراسة الحاسب بموجز تاريخى مهم يقدم فكرة عن بنيته و وظيفته ، ويلى ذلك التطرق الى الكفاءة العامة لنظام الحاسب والأعتبارات اللازم أتخاذها فى الحسبان لتحقيق التوازن بين موارد الحاسب .

2.1 تاريخ موجز للحاسبات

2.1.1 الجيل الأول : الأنابيب المفرّغة

كان نظام ENIAC (Electronic Numerical Integrator And Computer) المصمم والمُبني بجامعة ولاية بنسلفانيا أول حاسب رقمي الكتروني للأغراض العامة في العالم . وكان هذا المشروع أستجابة لأحتياجات الجيش الأمريكي خلال الحرب العالمية الثانية حيث كان مختبر أبحاث المقذوفات بالجيش يواجه صعوبة في توريد جداول مسارات المقذوفات بشكل دقيق وضمن إطار زمني معقول رغم عمل أكثر من 200 شخص بهذا المختبر ، و يستغرق إعداد الجداول للسلاح الواحد من الشخص الواحد عدة ساعات وقد تصل المدة لأيام.

أقترح "جون ماكلي" و "إيكرت جون" بناء جهاز حاسب للأغراض العامة باستخدام تقنية الأنابيب المفرّغة لتطبيقات مختبر أبحاث المقذوفات ، وفي عام 1943 بدأ العمل على نظام (ENIAC) وكانت النتيجة جهاز هائل يزن 30 طنا ويحتل 150 متراً مربع من المساحة والتي تضم أكثر من 18000 أنبوب مفرّغ .

ويستهلك عند التشغيل 140 كيلوواط من الطاقة وقادراً على إجراء 5000 عملية جمع في الثانية الواحدة ، وكان بناء (ENIAC) جهازاً عشرياً بدلاً من ثنائياً ، وهذا يعنى ان إجراء العمليات الحسابية يتم بالنظام العشري للارقام . وتألفت ذاكرته من 20 مسجل "مجمع" كل واحد منها قادر على حفظ رقم عشري مكون من 10 خانات . أما العيب الرئيسي فى حاسب (ENIAC) هو أنه كان لابد من برمجته يدويا عن طريق ضبط المفاتيح وتوصيل ونزع الأسلاك ، وقد تم الأنتهاء من انجازه في عام 1946 متأخراً عن الأستخدام فى المجهود الحربى ، وكانت المهمة الأولى له تنفيذ سلسلة من العمليات الحسابية المعقدة التي كانت تستخدم للمساعدة في تحديد الجدوى من القنبلة الهيدروجينية ، وهذا يبين أنه كان جهاز حاسب للأغراض العامة وجرى إستعماله فى أمر مخالف للغرض المصمم من أجله .

نموذج فون نيومان لنظام الحاسب الآلي

كانت برمجة آلالات الحاسبة الألكترونية الأولى مرادفة لتوصيل الأسلاك بالمقابس ، فلا توجد بنية طبقية ولذلك كانت برمجة الحاسب عملاً يغلب عليه طابع الهندسة الكهربائية وكتمرين على تصميم خوارزمية ما . وقبل أن ينهي كل من "جون موكلي" و "جون ايكرت" عملهم على بناء نظام الحاسب "اينياك" (ENIAC) تصورا طريقة أسهل لتغيير سلوك عمل هذه الآلة الحاسبة ، واعتقدا أن وحدات ذاكرة على شكل خطوط التأخير الزئبقية يمكن أن توفر وسيلة لتخزين تعليمات البرنامج ، وهذا من شأنه وضع حد نهائى للملل الناتج من تغيير الأسلاك الكهربائية للنظام في كل مرة يكون لديهم مشكلة جديدة لحلها أو قديمة لتصحيحها. موكلي وايكرت وثقا فكرتهما كمقترح لبناء النظام التالى للحاسب الخاصة بهما

وهو EDVAC - (Electronic Discrete Variable Computer) ، ولكن في أثناء إنشغالهما فى المشروع السرى لتصميم وبناء نظام (ENIAC) خلال الحرب العالمية الثانية لم يتمكن موكلي وايكرت من نشر فكرتهما على الفور . أحد الذين يعملون في محيط مشروع (ENIAC) كان عالم الرياضيات المجري الشهير جون فون نيومان ، وبعد قراءته لاقتراح ماوكلي وايكرت لفكرة (EDVAC) نشر فون نيومان هذه الفكرة والمعروفة بإسم **مفهوم تخزين البرنامج** ، وبذلك فعلياً هو الذى أوصل هذا المفهوم وقيد التاريخ هذا الأختراع له ، وكافة أنظمة الحاسب المخزنة للبرنامج تعرف بأنظمة فون نيومان وتستخدم معمارية فون نيومان . وعلى الرغم من أننا مضطرون لتقييد القول أن أنظمة الحاسب المخزنة للبرنامج تستخدم معمارية فون نيومان ولكن لا نفعل ذلك دون الإشادة المناسبة لمخترعها الحقيقي : جون موكلي وجون ايكرت . والنسخ الحديثة من الأنظمة ذات البنية المخزنة للبرنامج تستوفى على الأقل الخصائص التالية :

- تتكون من ثلاثة أنظمة مادية : وحدة المعالجة المركزية (CPU) مكونة من وحدة التحكم و وحدة الحساب والمنطق (ALU) ومسجلات (وحدات تخزين صغيرة) وعداد البرنامج ، ونظام الذاكرة الرئيسية التي تحفظ البرامج التي تتحكم في عمل الحاسب ، واخيراً نظام الإدخال/الإخراج .
- إمكانية إجراء المعالجة للتعليمات تسلسلياً .
- **تحتوي على مسار واحد - إما مادياً أو منطقيا - مابين ذاكرة النظام الرئيسة و وحدة التحكم فى وحدة المعالجة المركزية ، وهذا أجبر على التناوب مابين دورتى التعليمة والتنفيذ ، وغالبا ما يشار إلى هذا المسار الواحد بإسم عنق الزجاجة لنظام فون نيومان .**

ويبين الشكل (2.1) كيفية عمل هذه الميزات معاً في أنظمة الحاسب الحديثة.
ولاحظ أن النظام المبين في الشكل يمرر كل مدخلاته ومخرجاته من خلال وحدة الحساب والمنطق (في الواقع، فإنه يمررها عبر مسجل المجمع (Acc) والذي يعتبر جزءاً من وحدة الحساب والمنطق)، وهذه المعمارية تنفذ البرامج بما يعرف بإسم دورة فون نيومان للتنفيذ (وتسمى أيضا بدورة جلب – فك – تنفيذ) والتي تصف كيفية عمل المعالج.

الشكل (2.1) – معمارية فون نيومن لنظام الحاسب الآلي

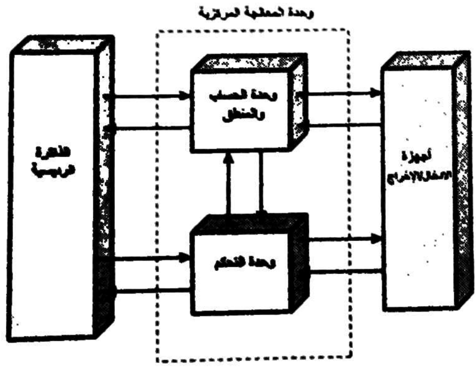
ويمكن سرد تكرار واحد للدورة على النحو التالي:

1. تجلب وحدة التحكم تعليمة البرنامج التالية من الذاكرة، وذلك بإستخدام عداد برنامج لتحديد مكان موقع التعليمة.
2. يتم فك شفرة التعليمة إلى لغة تفهمها وحدة الحساب والمنطق.

3. يتم جلب المعاملات من الذاكرة إذا تطلب تنفيذ التعليمة ذلك وتوضع في مسجلات داخل وحدة المعالجة المركزية.
4. وحدة الحساب والمنطق تنفذ التعليمة وتضع النتائج في مسجلات أو في الذاكرة.

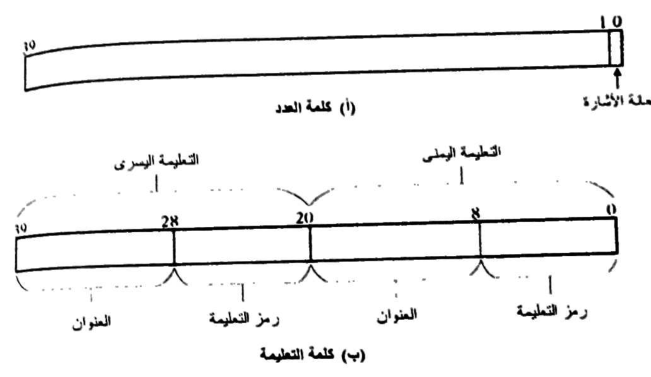
وبناء على فكرته بدأ فون نيومن في عام 1946 في تصميم حاسب جديد مُخزن للبرنامج في معهد برنستون للدراسات المتقدمة، ويشار إليه بإسم (IAS) (Institute of Advanced Studies)، والشكل (2.2) يبين الهيكل العام للحاسب (IAS) الذي يتكون من:

- الذاكرة الرئيسية والتي تخزن كل من البيانات والتعليمات.
- وحدة الحساب والمنطق وتعمل على البيانات الثنائية.
- وحدة التحكم، تفسر التعليمات التي في الذاكرة وتنفذها.
- وحدات الإدخال والإخراج وتديرها وحدة التحكم.



الشكل (2.2) – البنية المبسطة لنظام الحاسب (IAS)

مع إستثناءات نادرة ، فكل الحاسبات اليوم لديها نفس هذا الهيكل العام والوظيفي، ولذلك يشار اليها بأنظمة فون نيومن . وتتكون ذاكرة نظام الحاسب (IAS) من 1000 موقع تخزين تُسمى "كلمات" كل منها بسعة 40 عدد ثنائي (خانة) ، ويتم بها تخزين البيانات والتعليمات. ويتم تمثيل الأعداد على شكل ثنائي ، والتعليمة كرمز ثنائى ، ويوضح الشكل (2.3) هذه التنسيقات .



الشكل (2.3) – تنسيقات ذاكرة نظام الحاسب (IAS)

يتم تمثيل كل عدد بواسطة خانة للإشارة و 39 خانة للقيمة. وقد تحتوي الكلمة أيضا على تعليمتين لكل منهما 20 خانة ، وتتكون كل تعليمة من 8 خانات لشفرة التشغيل (رمز العملية) التى تحدد العملية التي يتعين القيام بها ، و 12 خانة تشير الى عنوان في الذاكرة (مرقمة من 0 الى 999).

تقوم وحدة التحكم بتشغيل نظام الحاسب (IAS) وذلك بجلب التعليمات من الذاكرة وتنفيذها واحدة تلو الأخرى . ولتفسير هذا يبين الشكل (2.4) رسماً تخطيطيا للبنية المفصلة للحاسب (IAS) ، والذى يكشف أن كلا من وحدة التحكم و وحدة

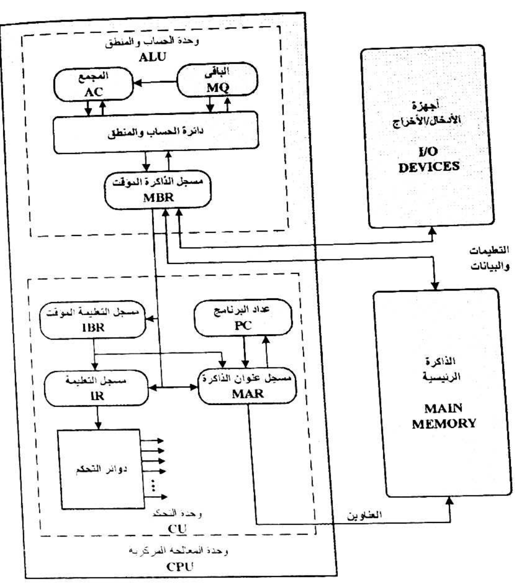
الحساب والمنطق تحتوي على مواقع للتخزين تُسمى **مسجلات** تعرّف على النحو التالي :

- مسجل الذاكرة المؤقت (MBR) : يحتوي على الكلمة التى سيتم تخزينها في الذاكرة أو إرسالها إلى وحدة الإدخال/الإخراج ، أو يُستخدم للحصول على كلمة من الذاكرة أو من وحدة الإدخال/الإخراج.
- مسجل عنوان الذاكرة (MAR) : يُحدد عنواناً في الذاكرة للكلمة المراد كتابتها أو قراءتها من و الى مسجل الذاكرة المؤقت (MBR).
- مسجل التعليمة (IR) : يحتوي على 8 خانات لشفرة تشغيل التعليمة (رمز العملية) الجارى تنفيذها.
- مسجل التعليمة المؤقت (IBR) : يستخدم كمخزن مؤقت للجزء الأيمن من كلمة الذاكرة المحتوية على التعليمة.
- عداد البرنامج (PC) : يحتوي على العنوان التالي لزوج التعليمات المراد جلبها من الذاكرة.
- المجمع (AC) والباقي (MQ) : يستخدم كمخزن مؤقت للمعاملات ونتائج عمليات وحدة الحساب والمنطق ، فعلى سبيل المثال نتيجة ضرب رقمين من 40 خانة هو رقم من 80 خانة ، يتم تخزين 40 خانة الأعلى فى المجمع (AC) ، و 40 خانة الأدنى فى الباقي (MQ).

يعمل الحاسب (IAS) بتكرار إجراء دورة التعليمة ، وتتكون دورة كل تعليمة من دورتين جزئيتين .

تُحمل خلال دورة الجلب شفرة تشغيل التعليمة التالية في مسجل التعليمة ، ويكون عنوانها مُحملا في مسجل عنوان الذاكرة ، ويمكن أن تؤخذ هذه التعليمة من

مسجل التعليمة المؤقت ، أو يمكن الحصول عليها من الذاكرة عن طريق تحميل كلمة فى مسجل الذاكرة المؤقت ، ومن ثم إلى المسجل المؤقت للتعليمة ومسجل التعليمة وصولا إلى مسجل عنوان الذاكرة لحفظ الجزء الخاص بالعنوان . ويتم اجراء دورة التنفيذ عندما يكون رمز التعليمة فى مسجل التعليمة . وتفسر دائرة التحكم رمز التعليمة وتنفذه عن طريق إرسال إشارات التحكم المناسبة لهذه التعليمة والتى تسبب نقل بيانات ، أو عملية يتعين على وحدة الحساب والمنطق القيام بها.



الشكل (2.4) – البنية التفصيلية لنظام الحاسب (IAS)

ويضم الحاسب (IAS) طقم بعدد 21 تعليمة ، **يمكن تصنيفها على النحو التالي:**

- نقل البيانات : نقل البيانات بين الذاكرة و مسجلات وحدة الحساب والمنطق ، أو بين مسجلين فى وحدة الحساب والمنطق.
- تفرع غير مشروط : عادةً وحدة التحكم تنفذ التعليمات من الذاكرة تسلسلياً ، ويمكن تغيير هذا التسلسل بواسطة تعليمة تفرع والتي تُمكِّن من إجراء عمليات التكرار.
- تفرع مشروط : يمكن جعل التفرع يعتمد على شرط ما ، وبالتالي يسمح باتخاذ قرار ما.
- حسابية : وهى العمليات التي تقوم بها وحدة الحساب والمنطق .
- تعديل العنوان : يسمح بحساب العناوين فى وحدة الحساب والمنطق ومن تم إدراجها في تعليمات مخزنة في الذاكرة ، ويسمح هذا للبرنامج بمرونة كبيرة فى العنونة.

في عام 1947 صنعت شركة إيكرت-ماكلي للحاسبات اول جهاز لها بنجاح وهو **يونيفاك-1** (UNIVAC-I) - (**Uni**versal **A**utomatic **C**omputer)، وكان هذا أول جهاز حاسب تجاري ناجح و معداً للتطبيقات العلمية والتجارية على حد سواء.

وأول وثيقة تصف عمل الجهاز تضمنت حساب جبر المصفوفات ، والمشاكل الإحصائية ، وفواتير الأقساط لشركات التأمين ، والمشاكل اللوجستية وذلك كعينة من المهام التي يمكنه أن يقوم بها.

2.1.2 الجيل الثاني : الترانزستورات

تم أختراع الترانزستور في مختبرات "بل" في عام 1947 ، وبحلول عام 1950 انطلقت الثورة الألكترونية ، إلا أن أجهزة الحاسب الترانزستورية لم تظهر تجاريا حتى أواخر الخمسينيات من القرن المنصرم . وتميز الجيل الثاني بظهور شركة المعدات الرقمية (Digital Equipment Corporation) أو أختصاراً (DEC) ، والتى تأسست في عام 1957 وقدمت في نفس العام حاسبها الأول (PDP-1) . وقدمت آي بي إم (IBM) السلسلة 700 من أنظمة الحاسب في عام 1952 ومن ثم تطورت الى السلسلة 7000 في عام 1964 حيث شكل هذا التطور نموذجا فى أنتاج الحاسبات وأظهرت الأعضاء المتتالية فى هذه السلسلة زيادة في الأداء، وفى القدرة ، و/أو نقص فى التكلفة.

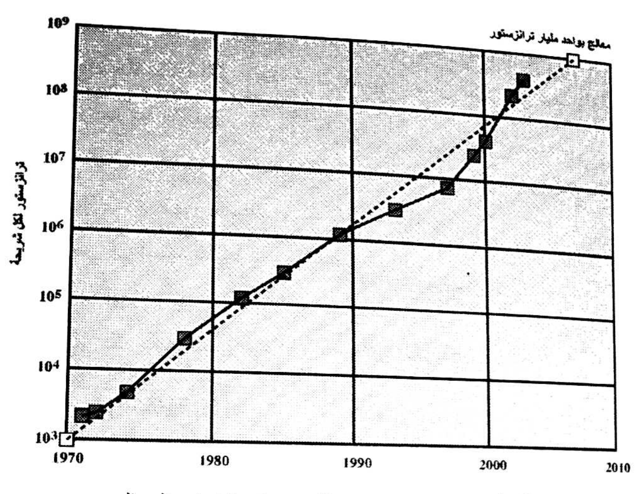
وأما حجم الذاكرة الرئيسية فهى عبارة عن مضاعفات لكلمة من 36 خانة بحيث زادت من 2 كيلو الى 32 كيلو كلمة (كيلو = 2^{10}) ، وفي حين إن وقت الوصول إلى كلمة فى الذاكرة وزمن دورة الذاكرة أنخفض من 30 ثانية إلى 1.4 ثانية ، كذلك أرتفع عدد شفرات التشغيل (رموز التعليمات) من 24 إلى 185 شفرة.

2.1.3 الجيل الثالث : الدوائر المتكاملة

في عام 1958 بدأ عصر الألكترونيات الدقيقة بأختراع الدوائر المتكاملة التي تُميز الجيل الثالث من أجهزة الحاسبات . في البداية لم يمكن على نحو موثوق تصنيع و تعبئة سوى عدد قليل من البوابات أو خلايا الذاكرة على شريحة الكترونية واحدة ويشار إلي هذه التقنية بمايسمى (SSI).

ومع مرور الوقت وتقدم تقنية تصنيع الألكترونيات أصبح من الممكن تعبئة عناصر أكثر فأكثر من الدوائر المنطقية على نفس الشريحة ويتضح هذا النمو

في الكثافة (زيادة فى عدد العناصر مع عدم تغيير فى حجم الشريحة) و الشكل (2.5) يوضح معدل أزدياد عدد الترنزستورات فى الشريحة مع الزمن بحيث أصبح خطياً - تقريبا.



الشكل (2.5) – نمو عدد الترنزستورات فى المعالج

يعكس هذا الشكل قانون "مــور" الشهير الذي طُرح من قبل "**غوردون مور**" في عام 1965 حيث لاحظ مور أن عدد الترانزستورات التي يمكن وضعها على شريحة واحدة يتضاعف كل عام ، وصح توقعه إذ إن هذه الوتيرة أستمرت حتى المستقبل القريب . وترتب على قانون **مور** النتائج التالية :

1. **تكلفة الشريحة ظلت دون تغيير تقريبا خلال هذه الفترة من النمو السريع في الكثافة ، وهذا يعني أن تكلفة دوائر المنطق ودوائر الذاكرة قد أنخفضت بمعدل كبير وهذا ساهم فى أستقرار تكلفة الحاسبات مع تطور كبير فى قدراتها و أدائها .**
2. **بما أن الدوائر المنطقية** ودوائر الذاكرة توضع قريبة من بعضها البعض على **شرائح معبأة بكثافة أكثر** ، فأنه قد تم تقصير المسار الكهربائي فيما بينها مما **زاد من سرعة التشغيل .**
3. صار الحاسب أصغر مما جعله أكثر ملائمة للوضع في مجموعات متنوعة **من البيئات**، و ظهور أصناف و أشكال متعدد له.
4. **أنخفاض في** أستهلاك الطاقة ومتطلبات التبريد.
5. الروابط فى الدوائر المتكاملة أكثر موثوقية من وصلات اللحام ، ومع دوائر **أكثر في الشريحة** ينقص عدد الروابط مابين الشرائح.

نظام حاسب (IBM/360)

بحلول عام 1964 أحكمت (IBM) من قبضتها على سوق الحاسبات مع **عائلة 7000** من حاسباتها ، وكان مفهوم العائلة المكون من حاسبات متوافقة **جديداً** وناجحاً للغاية بحيث يمكن للزبون ذو المتطلبات المتواضعة والميزانية **المحدودة** من البدء بنموذج 30 منها والغير مكلّف نسبيا ، ثم وفي وقت لاحق إذا **زادت احتياجاته** فمن الممكن الترقية لحاسب أسرع بذاكرة أكبر من دون التضحية **بالبرمجيات** المستخدمة ، والخصائص المميزة للعائلة هي :

1. **طقم تعليمات** متشابهة أو متطابقة .
2. **نظام التشغيل** متشابه أو متطابق .

3. **زيادة فى السرعة .**
4. **عدد متزايد من منافذ الإدخال/الإخراج .**
5. **زيادة حجم الذاكرة .**
6. **زيادة في التكاليف .**

2.1.4 الأجيال الاخيرة : الدوائر المتكاملة الفائقة

مع تسارع وتيرة التقنية وأرتفاع معدل إدخال منتجات جديدة ، **وزيادة أهمية** البرمجيات والإتصالات فضلاً عن أهمية الأجهزة **صار التصنيف بالأجيال أقل** وضوحا وذا مغزى أقل ، فمنذ السبعينيات **تطورت التطبيقات التجارية تطوراً** جديدا أدى إلى تغيير كبير فى الحاسبات ، ولا تزال **نتائج هذه التغييرات مستمرة .** ونذكر في هذا الجزء اثنين من أهم هذه النتائج .

الذاكرة الالكترونية

كان أول تطبيق لتقنية الدوائر المتكاملة فى الحاسبات هو **بناء الذاكرة ، ففي** الخمسينيات و الستينيات من القرن المنصرم كان معظم **ذاكرة الحاسب الأساسية** مغناطيسية ، وكانت الذاكرة المغناطيسية النواة سريعة إلى حد ما ؛ **إذ تستغرق** أقل من مليون من الثانية لقراءة خانة مخزنة في الذاكرة ، **ولكنها كانت باهظة** الثمن ، كبيرة الحجم وتستخدم القراءة المتلفة .

في عام 1970 أنتجت شركة "فيرتشايلد" أول ذاكرة الكترونية **(شبه موصله)** واسعة نسبيا حيث يمكن لهذه الشريحة أن **تخزن 256 خانة من الذاكرة ، وكانت** غير متلفة وأسرع بكثير من الذاكرة المغناطيسية النواة . **ومع ذلك فإن التكلفة لكل** خانة في الذاكرة الألكترونية **أعلى من نظريتها المغناطيسية النواة .**

وفي عام 1974 أنخفض سعر كل خانة من الذاكرة الألكترونية (شبه موصله) لأقل من سعر كل خانة من الذاكرة المغناطيسية النواة ، وبعد ذلك كان هناك أنخفاض مستمر وسريع في تكلفة الذاكرة يرافقه زيادة في كثافة الذاكرة الفعلية . ومنذ عام 1970 تطورت الذاكرة الألكترونية (شبه موصلة) لتزيد عن 32 غيغا (2^{30}) خانة ثنائية على الشريحة الواحدة ، فكل جيل زاد أربعة أضعاف من كثافة التخزين عن الجيل الذى سبقه ويرافق هذا انخفاض فى التكلفة لكل خانة ، وأنخفاض فى زمن الوصول للكلمة فى الذاكرة.

المعالجات

بما أن كثافة العناصر على شرائح الذاكرة أستمرت في الارتفاع ، فإن كثافة العناصر على شرائح المعالج أستمرت فى الارتفاع كذلك . ومع مرور الوقت وُضعت عناصر أكثر وأكثر في كل شريحة مما قلل الحاجة لشرائح أقل فأقل لبناء معالج حاسب واحد.

كانت أنتل-4004 أول شريحة تحتوي على كافة مكونات وحدة المعالجة المركزية على شريحة واحدة : لقد خرج **المعالج الدقيق** للوجود . فالمعالج 4004 له القدرة على جمع عددين كل منهما مكون من 4 خانات ويمكن له إجراء عملية ضرب بأسلوب الجمع المتكرر فقط . وكانت الخطوة الكبرى التالية في تطور المعالجات في عام 1972 بواسطة أنتل-8008 ، وهو أول معالج دقيق ذو 8 خانات ، وكان اكثر تعقيدا بمرتين تقريبا من أنتل-4004 . وفي الوقت نفسه تقريبا بدأت المعالجات ذات 16 خانة بالظهور ، ومع ذلك لم يتحقق هذا الظهور حتى نهاية السبعينيات بخروج معالج أنتل-8086 للأغراض العامة ذو 16 خانة. وحدثت الخطوة التالية في هذا الأتجاه في عام 1981 عندما أنتجت كل من

مختبرات "**بل**" و "**هيوليت باكارد**" معالجات ذات 32 خانة على شريحة واحدة. وبعدها قدمت أنتل المعالج 80386 ذو 32 خانة الخاص بها في عام 1985 ، ثم تبعه مؤخراً المعالجات 64 خانة المتعددة النواة .

الجدول التالي (الجدول - 2.1) يوضح تطور معالجات **أنتل** و ما صاحبه من تغير و تطور فى مكونات و تنظيم المعالج وإمكانياته .

الجدول (2.1) – تطور معالجات أنتل

المعالج	4004	8088	80286	486TM DX	Pentium	Pentium 2	Pentium 4	Core 2 Duo	Core 2 Quad
سنة الانتاج	1971	1979	1982	1989	1993	1997	2000	2006	2008
سرعة النبضة	108kHz	5MHz, 8MHz	6MHz-12.5MHz	25MHz-50MHz	60MHz-166MHz	200MHz-300MHz	1.3-1.8 GHz	1.06-1.2 GHz	3 GHz
عرض الناقل	4 bits	8 bits	16 bits	32 bits	32 bits	64 bits	64 bits	64 bits	64 bits
عدد الترانزستورات	2,300	29,000	134,000	1.2 million	3.1 million	7.5 million	42 million	167 million	820 million
حجم الترانزستور	10 *um*	6 *um*	1.5 *um*	0.8-1 *um*	0.8 *um*	0.35 *um*	180 *nm*	65 *nm*	45 *nm*
الذاكرة القابلة للعنونة	640 Bytes	1 MB	16 MB	4 GB	4 GB	64 GB	64 GB	64 GB	64 GB
الذاكرة الظاهرية	-	-	1 GB	64 TB	64 TB	64 TB	64 TB	64 TB	64 TB
الذاكرة السريعة	-	-	-	8 kB	8 kB	512 kB L2	256 kB L2	2 MB L2	6 MB L2

التصميم من أجل الأداء

سنة بعد سنة تناقصت تكلفة أنظمة الحاسب بشكل كبير فى حين أستمر أداء وقدرة **تلك النظم في الأرتفاع بنفس الوتيرة حتى أصبحت الحواسيب محطات عمل متطورة لتطبيقات شتى .**

تدعم محطات العمل الحاسوبية التطبيقات الهندسية والعلمية المتطورة للغاية فضلا عن نظم المحاكاة التى لها القدرة على دعم تطبيقات معالجة الصور والفيديو، ويضاف إلى ذلك أن الأعمال التجارية تعتمد بشكل متزايد على خوادم قوية لمعالجة المعاملات وتجهيز قواعد بيانات واسعة النطاق لدعم شبكات العميل/الخادم التي حلت محل الحاسبات المركزية الضخمة القديمة . وتسلط النقاط التالية الضوء على بعض العوامل التى تؤدى الى تصميم أنظمة حاسب ذات آداء عالى .

2.2.1 سرعة المعالج

لن تصل سرعة المعالج الى الحد الأقصى لإمكانياته ما لم يتم تغذيته بدفق مستمر من الشغل للقيام به على شكل تعليمات حاسوبية ، لذا يجب على مصممي المعالج الخروج بتقنيات أكثر تطورا من آي وقت مضى لتغذية هذا الوحش . ومن بين التقنيات المعتمدة فى تصميم كيان المعالجات المعاصرة مايلى :

1- **التنبؤ بالتفرع : يتطلع المعالج الى الأمام عند جلب التعليمة التالية من الذاكرة ، ويتنبأ بأى تفرع (التعليمة التي سوف تنفذ ليست التي تلى المنفذة) أو مجموعة التعليمات التى من المرجح أن يتم معالجتها تالياً .**

2- **تحليل تدفق البيانات : يحلل المعالج التعليمات المكونة للبرنامج لتحديد أي من التعليمات تعتمد نتائجها أو بياناتها على بعضها البعض و ذلك لإنشاء خطة مُثلى لتنفيذ التعليمات .**

3- التنفيذ المتضارب : باستخدام التنبؤ بالتفرع وتحليل تدفق البيانات تنفِّذ بعض المعالجات التعليمات تخمينيا قبل ظهورها الفعلي في تسلسل تنفيذ البرنامج، وتخزِّن النتائج في مواقع مؤقتة.

2.2.2 توازن الأداء

بما أن قوة المعالج تتقدم بسرعة فائقة ، فأننا نجد أن باقى المكونات المهمة الأخرى للحاسب لم تواكب هذا التقدم ، ونتيجة لذلك فأننا فى حاجة للبحث عن توازن فى الأداء : تعديل فى تنظيم وهندسة معمارية الحاسب للتعويض عن عدم التوافق بين القدرات المختلفة للمكونات. ولا يوجد مكان للمشكلة الناجمة عن عدم التطابق هذا أكثر أهمية من العلاقة مابين المعالج والذاكرة الرئيسية (انظر الشكل 2.6) ، فإذا فشلت الذاكرة أو الناقل فى مواكبة متطلبات المعالج الفورية سيتعطل المعالج ويدخل في حالة انتظار مما يضيع وقتا ثمينا للمعالجة ، وهناك عدة طرق لحل هذه المعضلة ، منها :

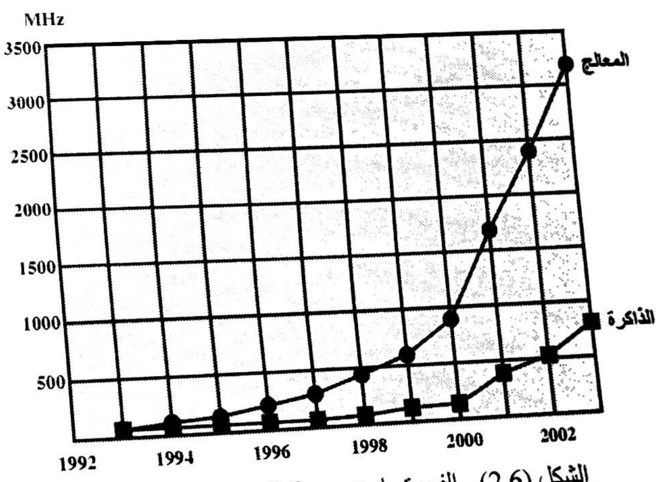
1- زيادة عدد الخانات التي يتم استردادها (جلبها) في وقت واحد عن طريق جعل الذاكرة "أوسع" بدلا من "أعمق" ، وبإستخدام ناقل بيانات واسع المسارات .

2- تغيير رابط الذاكرة التفاعلية (DRAM) بجعله أكثر كفاءة من قبل بما في ذلك إستخدام نظام التخزين السريع (Cache) أو غيره على شرائح الذاكرة التفاعلية (DRAM).

3- الحد من وتيرة الوصول إلى الذاكرة من خلال زيادة إستخدام التخزين السريع (Cache) على نحو متزايد وفعال بين المعالج والذاكرة الرئيسية. ويشمل هذا إدراج وحدة أو أكثر من التخزين السريع على شريحة المعالج و خارجها على مقربة منها.

4- زيادة عرض نطاق التواصل بين المعالجات والذاكرة عن طريق إستخدام ناقلات عالية السرعة ، وبإستخدام تسلسل هرمي من الناقلات لعزل وهيكلة تدفق البيانات .

5- مجال آخر من المجالات التي ركز عليه التصميم هو التعامل مع أجهزة الإدخال/الإخراج . كلما أصبحت أجهزة الحاسب أسرع وأكثر قدرة خرجت تطبيقات أكثر تطوراً وتدعم إستخدام أجهزة طرفية بمتطلبات إدخال/إخراج كثيفة .



الشكل (2.6) - الفجوة مابين سرعة الذاكرة والمعالج

2.2.3 التحسينات في تنظيم وعمارة الشريحة

لازال صراع المصممين قائما على التحدي المتمثل في تحقيق التوازن مابين أداء المعالج مع الذاكرة الرئيسية وغيرها من مكونات الحاسب والحاجة إلى زيادة سرعة المعالج . هناك ثلاثة أساليب لتحقيق زيادة فى سرعة المعالج :

1- زيادة سرعة العتاد المادى للمعالج . وتتم هذه الزيادة أساسا بتقليص حجم البوابات المنطقية على شريحة المعالج بحيث يمكن أن تكون بكثافة أكثر، وكذلك بزيادة معدل النبضة .

2- زيادة حجم وسرعة الذاكرة السريعة التي تتوسط مابين المعالج والذاكرة الرئيسية من خلال تخصيص جزء من شريحة المعالج نفسه إلى الذاكرة السريعة .

3- إجراء تغييرات على تنظيم المعالج ومعماريته بحيث تزيد من سرعة التنفيذ الفعال للتعليمة وعادة ماينطوي هذا على إستخدام تقنية التنفيذ المتوازي بشكل أو بآخر.

تقليديا ، العامل المهيمن على تحقيق مكاسب في الأداء هو الزيادة في السرعة وكثافة المنطق (عدد الدوائر المنطقية على الشريحة) ، وقد صحب ذلك بروز عدد من المعضلات أهمها :

1- الطاقة : زيادة فى أستهلاك الطاقة مع صعوبة فى تبديد الحرارة المتولدة.

2- التأخير : التأخير الناتج من الظاهرة الكهرومغناطيسية مابين المكثف و المقاومة ، فكلما تقلص حجم مكونات الشريحة تزداد رقة الاسلاك مما يزيد من مقاومتها الكهربية وكذلك تقترب الأسلاك من بعضها اكثر مما يزيد من سعويتها (الكثافة الكهربائية مابين الأسلاك) .

3- تأخر الذاكرة : سرعة الذاكرة تتخلف عن سرعة المعالج .

مع أواخر 1980 تم استخدم استراتيجيتين رئيسيتين لزيادة الأداء هما :

أولا ، كان هناك زيادة في قدرة الذاكرة السريعة ، ويوجد الآن عادة اثنين أو ثلاثة مستويات من وحدات الذاكرة السريعة بين المعالج والذاكرة الرئيسية.

ثانيا ، أصبح منطق تنفيذ التعليمات داخل المعالج معقداً بشكل متزايد حيث مكن من تنفيذ التعليمات بشكل متوازى داخل المعالج .

2.3 تقييم الأداء

لتقييم الكيان المادى للمعالج ولوضع المتطلبات لأنظمة جديدة ، يجب الأخذ فى الاعتبار الأداء حيث هو أحد المعايير الأساسية جنبا إلى جنب مع التكلفة والحجم و الموثوقية والأمن وكذلك في بعض الحالات أستهلاك الطاقة.

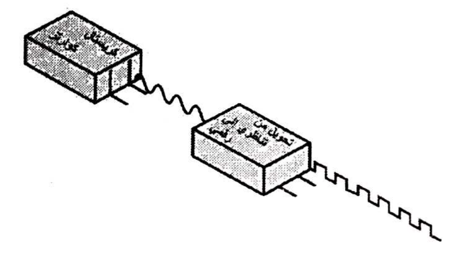
من الصعب إجراء مقارنة ذات مغزى فى الأداء بين معالجات مختلفة و حتى بين عائلة واحدة من المعالجات ، فالسرعة الاساسية هى أقل أهمية من كيفية عمل المعالج عند تنفيذ تطبيق معين . وللأسف فأن أداء التطبيقات لا تعتمد فقط على السرعة الاساسية للمعالج ، ولكن على طقم التعليمات ، وأختيار لغة التنفيذ، وكفاءة المترجم ، وخوارزمية حل المشكلة ، ومهارة البرمجة لتنفيذ تطبيق ما.

ففى هذا القسم نتطرق الى بعض المقاييس التقليدية لسرعة المعالج ، ثم ندرس النهج الأكثر شيوعا لتقييم أداء المعالج و نظام الحاسب .

2.3.1 سرعة النبضة و معدل التعليمات

ساعة النظام : إن **العمليات التي يقوم بها المعالج من مثل جلب تعليمة ، أو فك** شفرتها ، أو تنفيذ عملية حسابية ، أو ... الخ **تحكمها نبضة ساعة النظام . وتبدأ** عادة جميع العمليات مع نبضة الساعة ، وهكذا - **على المستوى الأساسي – فأن** سرعة المعالج يحددها تردد النبضة التي تنتجها الساعة ، **وتقاس بدورة في الثانية** أو هرتز (Hz).

وتنتَج إشارات الساعة عادة من كريستال الكوارتز، والتي تولد **إشارة موجة ثابتة** عند تغذيتها بالطاقة ، ويتم تحويل هذه الاشارة إلى نبضة جهد رقمي **بحيث يتم** توفيرها في تدفق مستمر لدوائر المعالج (الشكل (2.7)).



الشكل (2.7) – نبضة ساعة النظام

فعلى سبيل المثال ، فأن المعالج بواحد غيغاهرتز يتلقى 1,000,000,000 **نبضة** في الثانية الواحدة (1 بليون) حيث يعرف معدل النبضات **بمعدل الساعة أو سرعة** الساعة . ويشار الى النبضة الواحدة للساعة **بدورة الساعة ، والزمن مابين** النبضات **بزمن الدورة**.

ينطوي تنفيذ التعليمة على عدد من الخطوات المنفصلة مثل جلب التعليمة من الذاكرة ، فك الأجزاء المختلفة من التعليمة ، تحميل وتخزين البيانات ، إجراء العمليات الحسابية والمنطقية . وبالتالي ، فإن معظم التعليمات فى معظم المعالجات تتطلب نبضات متعددة للانتهاء ، وقد تستغرق بعض التعليمات بضع نبضات والبعض الأخر يتطلب العشرات . وهكذا ، فإن المقارنة المباشرة بسرعة الساعة لمعالجات مختلفة لا تعكس الصورة الكاملة للأداء.

معدل تنفيذ التعليمات : يتم تشغيل المعالج بواسطة ساعة بتردد ثابت - (f) أو بالمقابل ، دورة زمنية ثابتة (τ) ، حيث : $\tau = 1/f$. وبتحديد عدد التعليمات (I_c) لبرنامج ما ، وهو عدد تعليمات المعالج المنفذة لذلك البرنامج حتى يتم أنجازه الى النهاية أو لفاصل زمني محدد ، ومع ملاحظة أن هذا هو عدد التعليمات المنفذة وليس عدد التعليمات المذكورة في البرنامج . والمهم هنا هو **متوسط النبضات لكل تعليمة (CPI)** للبرنامج ، فإذا كانت كل التعليمات تتطلب نفس العدد من النبضات ، ففى هذه الحالة ستكون قيمة متوسط النبضات لكل تعليمة بالمعالج ثابتة ، ولكن ، فى أى معالج تختلف عدد النبضات المطلوبة باختلاف نوع التعليمة، مثل تخزين ، تحميل ، حساب ، تفرع ، ... الخ . فإذا كانت (CPI_i) هى عدد النبضات المطلوبة لتنفيذ نوع معين من التعليمة (i) ، (I_i) هى عدد التعليمات المنفذة من النوع (i) لبرنامج ما . اذن يمكننا حساب المتوسط العام للنبضات لكل تعليمة لبرنامج ما على النحو التالي:

$$CPI = \frac{\sum_{i=1}^{n}(CPI_i \times I_i)}{I_c}$$

يمكن التعبير عن زمن المعالج (T) اللازم لتنفيذ برنامج معين هو :

$$T = I_c \times CPI \times \tau$$

يمكننا تحسين هذه الصيغة من خلال الأقرار بأنه أثناء تنفيذ تعليمة ما فإن جزء من العمل يتم من قبل المعالج ، وجزء من الزمن يستغرق لنقل كلمة من أو إلى الذاكرة . وفي الحالة الأخيرة ، فإن زمن النقل يعتمد على زمن دورة الذاكرة والتي قد تكون أكبر من دورة زمن المعالج . ويمكننا إعادة كتابة المعادلة السابقة كما يلى :

$$T = I_c \times [p + (m \times k)] \times \tau$$

حيث (p) هو عدد دورات المعالج اللازمة للفك وتنفيذ التعليمة ، (m) هو عدد التأشيرات المطلوبة للذاكرة لجلب التعليمة ، و (k) هو النسبة بين زمن دورة الذاكرة و زمن دورة المعالج . إن عوامل الأداء الخمسة المذكورة في المعادلة السابقة (m, k, I_c, p, τ) تتأثر بسمات النظام الأربع : تصميم طقم التعليمات ، و تقنية المترجم (مدى فعالية المترجم في إنتاج برنامج كفؤ بلغة الآلة من برنامج بلغة راقية المستوى)، وتنظيم المعالج ، والذاكرة السريعة والتسلسل الهرمي للذاكرة.

الجدول (2.2) هو مصفوفة توضح أحد أبعادها عوامل الأداء الخمسة والبعد الأخر يظهر سمات النظام الأربع . الخانة التى بها (X) في الجدول تشير إلى سمة النظام التي تؤثر على عامل الأداء.

الجدول (2.2) - تأثير عوامل الأداء على سمات نظام الحاسب الآلي

	I_c	p	m	k	τ
معمارية طقم التعليمات	X	X			
تقنية المترجم	X	X	X		
تصميم المعالج		X			X
هيكلية الذاكرة الرئيسية و الذاكرة السريعة				X	X

مقياس شائع آخر لقياس أداء المعالج هو **معدل تنفيذ التعليمات** و يشار إليه بمليون تعليمة لكل ثانية واحدة (MIPS) ، و يمكن التعبير عن معدل تنفيذ التعليمات (MIPS) من حيث سرعة الساعة والمتوسط العام للنبضات لكل تعليمة (CPI) على النحو التالي : (معدل تنفيذ التعليمات)

$$(MIPS) - [\, I_c \,/\, (T \times 10^6)\,]$$

$$(MIPS) - [\, f \,/\, (CPI \times 10^6)\,]$$

على سبيل المثال ، تنفيذ برنامج ما نتج عنه تنفيذ 2 مليون تعليمة على معالج بسرعة 400 ميغاهيرتز ، ويتكون البرنامج من أربعة أنواع رئيسية من التعليمات، عدد أنواع التعليمات ونبضات كل تعليمة (CPI) لكل نوع من التعليمات معطى أدناه :

عدد نوع التعليمات	CPI	نوع التعليمات
%60	1	حسابي و منطقي
%18	2	تحميل/تخزين
%12	4	تفرع
%10	8	تأشير للذاكرة الرئيسية لعدم التوفر في الذاكرة السريعة

متوسط نبضات كل تعليمة عند تنفيذ البرنامج على معالج أحادي :

$$CPI = (0.1 \times 8) + (0.12 \times 4) + (0.18 \times 2) + (0.6 \times 1) = 2.24$$

في المقابل فإن معدل تنفيذ التعليمات هو :

$$MIPS = (400 \times 10^6) / (2.24 \times 10^6) - \sim 178 \,.$$

2.3.2 المعايير

أثبتت المقاييس مثل معدل تنفيذ التعليمات أنها غير كافية لتقييم أداء المعالجات، وذلك بسبب الأختلافات في طقم التعليمات ، ولذلك فإن معدل تنفيذ التعليمات ليس الوسيلة الصالحة لمقارنة أداء معماريات مختلفة . وعلاوة على ذلك فقد يكون اداء معالج معين على برنامج معين غير مفيد في تحديد الكيفية التي سوف يعمل بها المعالج على تطبيق من نوع مختلف جدا.

وبناء عليه ، فأنه في أواخر الثمانينات و في وقت مبكر من التسعينات أنتقل أهتمام الصناعة والأكاديميات إلى قياس أداء النظم باستخدام مجموعة من البرامج المعيارية ، ويمكن تشغيل نفس المجموعة من هذه البرامج على أجهزة مختلفة ومن تم مقارنة زمن تنفيذ كل منها وتقييم أداء النظام بناءً علي استجابته لها .

ومن الخصائص التي تميز البرامج المعيارية :

1. **مكتوبة بلغة راقية المستوى (مثلاً ++C) مما يجعلها قابلة للتنفيذ على أجهزة مختلفة .**
2. **تمثل نوعاً معيناً من نمط البرمجة مثل برمجة النظم أو البرمجة العددية أو البرمجة التجارية .**
3. **يمكن قياس أداءها بسهولة .**
4. **لديها شبكة توزيع واسعة .**

مصطلحات مهمة

مجمع	Accumulator (AC)
الدوائر المتكاملة	Integrated Circuit (IC)
تكامل دوائر بمقياس صغير	Small Scale Integration (SSI)
الذاكرة الرئيسية	Main Memory
مسجل عنوان الذاكرة	Memory Address Register (MAR)
مسجل الذاكرة المؤقت	Memory Buffer Register (MBR)
المعالج الدقيق	Microprocessor
متعدد النوة	Multicore
رمز العملية	Operation Code - Opcode
مُصنّع للمعدات الأصلية	(OEM) Original Equipment Manufacturer
وحدة تحكم البرنامج	Program Control Unit
حاسب مخزن للبرنامج	Stored Program Computer
كلمة	Word
وحدة الحساب والمنطق	Arithmetic and Logic Unit (ALU)
شريحة	Chip
دورة التنفيذ	Execute Cycle
دورة الجلب	Fetch Cycle
المسجل التعليمة المؤقت	Instruction Buffer Register (IBR)
دورة التعليمة	Instruction Cycle
مسجل التعليمة	Instruction Register (IR)
فئة/طقم التعليمات	Instruction Set
معمارية طقم التعليمات	Instruction Set Architecture (ISA)
خانة ثنائية	Bit
ثنائي	Binary
ذاكرة سريعة	Cache
عنوان	Address
عداد البرنامج	Program counter (PC)
ناقل النظام	System bus
تعليمة	Instruction
الذاكرة التفاعلية	Dynamic Random Access Memory (DRAM)
الذاكرة السريعة	Cache Memory
نبضات لكل تعليمة	Cycles per Instruction (CPI)
مليون تعليمة لكل ثانية واحدة	Millions of Instructions per Second (MIPS)

ثمان - 8 خانات ثنائية	Byte (B)
كيلو ثمان خانات ثنائية	Kilo Byte (KB-2^{10})
ميغا ثمان خانات ثنائية	Mega Byte (MB-2^{20})
غيغا ثمان خانات ثنائية	Giga Byte (GB-2^{30})
تيرا ثمان خانات ثنائية	Tera Byte (TB-2^{40})
المعايير	Benchmarks
شبه الموصلة	Semiconductor
ميكرو متر	Micro Meter (*um* – 10^{-6})
نانو متر	Nano Meter (*nm* – 10^{-9})
كيلو دورة في الثانية (هرتز)	K Hz (10^{3})
ميغا دورة في الثانية (هرتز)	M Hz (10^{6})
غيغا دورة في الثانية (هرتز)	G Hz (10^{9})

أسئلة للمراجعة

1. ما هي المكونات الرئيسية الأربعة لأى جهاز حاسب متعدد الأغراض حسب مفهوم نيومن؟
2. ماهى النتائج المترتبة على قانون مور؟
3. ماهى السمات المميزة للأجيال الأخيرة للحاسبات؟
4. ما هي التقنيات الحديثة المتبعة لزيادة سرعة المعالج الدقيق؟
5. ماهى الأستراتيجيات الحديثة للرفع من أداء المعالج؟
6. حسب فهمك ، ماهى أصناف انظمة الحاسب الآلي؟
7. الجامعة تريد إقتناء حاسب للشئون الأدارية فما هو الأفضل لها حاسب ذو سرعة معالجة عالية أم حاسب ذو قدرة تخزينية كبيرة وبدون الأخذ فى الأعتبار التكلفة ؟ (علل).
8. ماهى البرامج المعيارية و ماهى الخصائص المرغوبة فيها؟
9. أثناء التجول في متجر للحاسبات ، سمعت أحد العملاء يسأل البائع ما هو أسرع جهاز حاسب في المتجر يمكن شرائه ، رد البائع "اذا أنت تبحث عن ماكنتوش ، أسرع ماكنتوش لدينا يشتغل بسرعة ساعة 1.2 غيغا هرتز. إذا كنت تريد حقا أسرع معالج ، عليك شراء أنتل بنتيوم -4 لدينا بسرعة 2.4 غيغا هرتز بدلا من ذلك" .هل البائع على حق؟ ماذا ستقول لمساعدة هذا العميل؟
10. برنامج معيارى ينفذ على معالج بسرعة 200 ميغا هرتز ، البرنامج المنفذ يتضمن تنفيذ 200,000 تعليمة ، عدد التعليمات المنفذة و دورة الساعة لكل نوع من التعليمات كما يلى :

نوع التعليمة	عدد التعليمات	عدد الدورات لكل تعليمة
حسابي	90000	1
نقل بيانات	64000	2
حسابي بالنقطة العائمة	36000	3
تحكم	10000	2

أوجد متوسط (CPI) و معدل (MIPS) والزمن اللازم لتنفيذ البرنامج (T)؟

11. بأفتراض 3 معالجات مختلفة و بأطقم تعليمات مختلفة ، والمعالجات تشتغل بسرعة 200 ميغا هرتز ، تم تسجيل النتائج التالية لبرنامج منفذ عليها بعدد 5 مليون تعليمة :

نوع التعليمة	عدد التعليمات	عدد الدورات لكل تعليمة
المعالج-1 :		
حسابي	%41	7
تحميل وتخزين	%17	2
تفرع	%29	3
غيره	%13	1
المعالج-2 :		
حسابي	%41	1
تحميل وتخزين	%19	2
تفرع	%29	2
غيره	%13	1
المعالج-3 :		
حسابي	%41	2
تحميل وتخزين	%17	3
تفرع	%29	3
غيره	%13	1

أوجد متوسط (CPI) و معدل (MIPS) والزمن اللازم لتنفيذ البرنامج (T) على كل معالج ، و ماهو النوع من التعليمات الذى له تأثير مباشر فى الفرق بين أداء المعالجات ، و أى المعالجات افضل؟



الفصل الثالث

التركيب العام لنظام الحاسب الآلي

3- التركيب العام لنظام الحاسب : الوظائف والروابط

فى المستوى الأعلى ، يتكون نظام الحاسب من المكونات الأساسية المتمثلة في وحدة المعالجة المركزية والذاكرة و وحدات الإدخال/الإخراج وبحيث يمكن لكل مكون أن يتضمن عنصراً أو أكثر ، وترتبط هذه المكونات بشكل معين لتحقيق الوظيفة الأساسية للحاسب وهى تنفيذ البرامج ، وبالتالي يمكننا وصف نظام الحاسب عن طريق (1) وصف السلوك الخارجي لكل مكون – بمعنى إشارات البيانات والتحكم المتبادلة مع المكونات الأخرى ، (2) وصف هيكل الرابط البيني والضوابط اللازمة لإدارة و إستخدام بنية هذا الرابط.

إن وجهة النظر هذه لبنية و وظيفة الحاسب مهمة جداً لأنها تفسر وتعطى القدرة على فهم طبيعة الحاسب ، وبنفس القدر من الأهمية يمكن إستخدامها لفهم القضايا المعقدة لتقييم الأداء .

إن فهم هيكل و وظيفة نظام الحاسب يقدمان نظرة ثاقبة للأختناقات في النظام ، والمسارات البديلة ، وحجم الفشل في حالة فشل مكون ما ، ويسهل من إضافة تحسينات فى الأداء. ففي العديد من الحالات يتم الوفاء بمتطلبات أعلى لأداء نظام وقدرات التأمين من الفشل من خلال تغيير التصميم بدلا من مجرد زيادة سرعة وموثوقية المكونات الفردية للنظام .

يبدأ الفصل مع دراسة موجزة للمكونات الأساسية للحاسب ومتطلبات الربط الخاصة بها ، ثم يقدم نظرة وظيفية عامة عن الحاسب ، بعد ذلك يقدم دراسة عن

إستخدام النواقل فى ربط مكونات النظام ، والهياكل الأساسية المستخدمة لربط الأتصال البيني مابين مكونات الحاسب.

3.1 مكونات نظام الحاسب

تستند جميع تصاميم نظم الحاسب المعاصرة تقريبا على المفاهيم التي وضعها جون فون نيومان في معهد الدراسات المتقدمة بجامعة برينستون ، ويشار إلى هذا التصميم ببنية "فون نيومان" وهو يقوم على ثلاثة مفاهيم رئيسية هي:

- يتم تخزين البيانات والتعليمات في ذاكرة واحدة للقراءة والكتابة.
- محتويات هذه الذاكرة معنونة حسب الموقع دون أعتبار لنوع البيانات المخزنة في الموقع.
- التنفيذ يحدث بطريقة تسلسلية (ما لم يُعدل صراحةً) من تعليمة واحدة إلى التي تليها.

لقد تمت مناقشة الأسباب الكامنة وراء هذه المفاهيم في السابق ولكن يجدر تلخيصها هنا . الكيان المنطقى فى التصميم بالمنطق الرقمى عبارة عن مجموعة صغيرة من المكونات المنطقية الأساسية التي تدمج بطرق مختلفة لتخزن بيانات ثنائية وتؤدى عمليات حسابية ومنطقية على تلك البيانات ، فإذا كان المطلوب إجراء عملية حسابية خاصة يمكن بناء وحدة من مكونات منطقية تصمم خصيصا لإجراء هذا الحساب . ويمكن أن نجعل عملية ربط مكونات منطقية مختلفة في تكوين معين كشكل من أشكال البرمجة ، و"البرنامج" الناتج هو في شكل كيان مادى يسمى برنامج مادى (Hardwired Program).

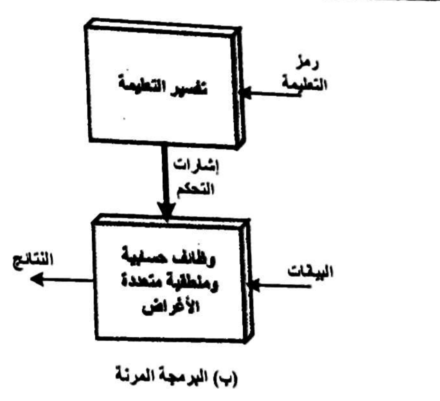
البديل لما سبق هو أن نفترض بناء وحدة ذات أغراض عامة للوظائف الحسابية والمنطقية ، وتؤدى هذه الوحدة وظائف مختلفة على البيانات إعتماداً على إشارات التحكم التى تُطبَق عليها (تدخل اليها) . ففي النموذج الأول من الكيان ذو التصميم المنطقى الخاص يقبل النظام بيانات ويعطي نتائج بعد معالجة البيانات المدخلة حسب تصميمه المنطقى (الشكل 3.1 – أ) . أما فى الكيان ذو التصميم المتعدد الأغراض يقبل النظام بيانات وأشارات تَحكم ويعطي نتائج بعد المعالجة التى تتم بنأ على اشارات التحكم ، وبهذا ، بدلا من تغيير تصميم الكيان لكل برنامج جديد يحتاج المبرمج لتغذية الكيان بإشارات تحكم جديدة لتغيير كيفية المعالجة .

كيف سيتم توليد إشارات التحكم ؟ ، الجواب بسيط ولكن متقن . فالبرنامج بأكمله هو في الواقع عبارة عن سلسلة من الخطوات ، وفي كل خطوة يتم تنفيذ بعض العمليات الحسابية أو المنطقية على بعض البيانات ، اذاً نحتاج فى كل خطوة إلى مجموعة جديدة من إشارات التحكم ويمكننا تخصيص رمز فريد لكل مجموعة ممكنة من إشارات التحكم ، ثم نضيف إلى الكيان المتعدد الأغراض جزء يمكنه من أن يقبل هذا الرمز ويولد إشارات التحكم حسب هذا الرمز ويرسلها الى وحدة ذات وظائف حسابية ومنطقية متعددة (الشكل 3.1 – ب).



(أ) البرمجة المادية

الشكل (3.1 – أ) - أسلوبى البرمجة : المادى والمرن



(ب) البرمجة المرنة

الشكل (3.1 – ب) - أسلوبى البرمجة : المادى والمرن

فالبرمجة الآن أكثر سهولة ، إذ بدلاً من تغيير الكيان المادى لكل برنامج جديد فإن كل ما يجب القيام به هو تقديم سلسلة جديدة من الرموز ، وكل رمز هو في الواقع تعليمة ، وجزء من النظام يفسر كل تعليمة ويولد إشارات التحكم الخاصة بها ، ولتمييز هذا الأسلوب الجديد من البرمجة يطلق على هذه السلسلة من الرموز أو التعليمات بالبرنامج المرن (Software).

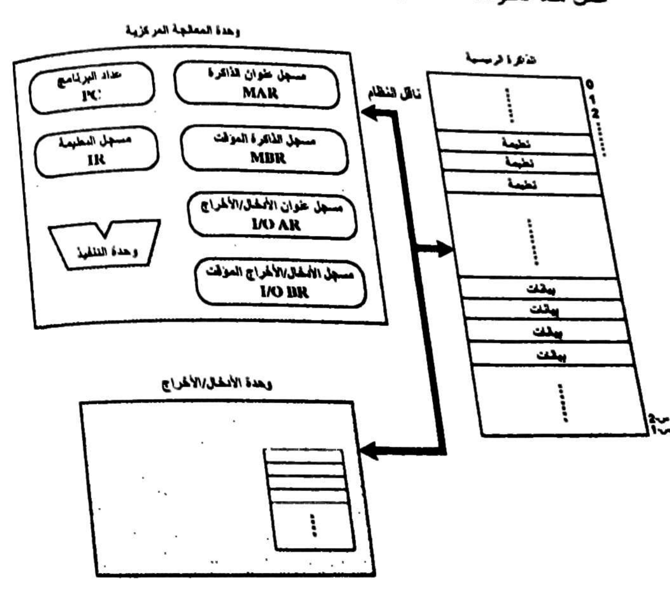
ويشير الشكل (3.1 – ب) للعنصرين الرئيسيين فى النظام : مفسر التعليمات و الوحدة المتعددة الأغراض للوظائف الحسابية والمنطقية ، هذان العنصران يشكلان وحدة المعالجة المركزية . ونحتاج إلى مكونات أخرى لإخراج نظام حاسب للعمل إذ يجب أدخال البيانات والتعليمات الى النظام ، ولهذا نحن بحاجة إلى وحدة للإدخال ، حيث تحتوي هذه الوحدة على المكونات الأساسية لقبول البيانات والتعليمات في شكل ما وتحويلها إلى نمط من الإشارات الداخلية القابلة

للأستخدام من قبل النظام . كذلك هناك حاجة إلى وسائل لإخراج النتائج ، وهذا يتم بواسطة وحدة الاخراج . و يشار الى الوحدتين كمكونات الإدخال/الاخراج . مطلوب مكون أخر ، فسوف تجلب وحدة الإدخال التعليمات والبيانات بالتتابع لكن البرنامج لا ينفذ دائما بالتتابع ، بل قد يكون هناك قفز (على سبيل المثال تعليمة الأنتقال في نظام الحاسب (IAS)) وبالإضافة الى ذلك قد يتطلب العمل على البيانات الوصول إلى أكثر من عنصر واحد في وقت واحد في تسلسل محدد سلفاً، وبالتالي يجب أن يكون هناك مكان للتخزين المؤقت لكل من التعليمات والبيانات ، ويسمى ذلك بوحدة الذاكرة أو الذاكرة الرئيسية لتمييز ها عن التخزين الخارجي أو الأجهزة الطرفية ، وأشار "فون نيومان" إلى أن نفس الذاكرة يمكن ان تستخدم لتخزين التعليمات والبيانات على حد سواء. ويوضح الشكل (3.2) هذه المكونات من المستوى الأعلى ، ويقدم التعاملات فيما بينها .

وحدة المعالجة المركزية تتبادل البيانات مع الذاكرة ، لهذا الغرض ، فهي عادة ماتستخدم أثنين من المسجلات الداخلية (بالنسبة الى وحدة المعالجة المركزية) ، مسجل عنوان ذاكرة (MAR) الذي يُحدد عنوان في الذاكرة للقراءة أو الكتابة التالية ، ومسجل الذاكرة المؤقت (MBR) الذي يحتوي على البيانات المراد كتابتها فى الذاكرة أو يتلقى البيانات المقروءة من الذاكرة ، بالإضافة الى ذلك مسجل عنوان الإدخال/الإخراج (I/OAR) الذى يحدد وحدة إدخال/إخراج معينة (عنوانها) ويستخدم المسجل المؤقت للإدخال/الاخراج (I/OBR) لتبادل البيانات بين وحدة الإدخال/الإخراج و وحدة المعالجة المركزية. وتتكون وحدة الذاكرة من مجموعة مواقع تحددها عناوين مرقمة بالتسلسل .

ويحتوي كل موقع على عدد من الثنائيات التي يمكن أن تفسر على أنها إما تعليمة أو بيانات . وتنقل وحدة الإدخال/الإخراج البيانات من الأجهزة الخارجية (الطرفيات) إلى وحدة المعالجة المركزية والذاكرة ، والعكس بالعكس (في الأتجاهين) ، وتحتوي على مخزن داخلي مؤقت لحفظ هذه البيانات مؤقتا الى حين إرسالها .

وبعد هذا التوضيح الوجيز للمكونات الرئيسية ، ننتقل إلى لمحة عامة عن كيفية عمل هذه المكونات معا لتنفيذ برنامج .



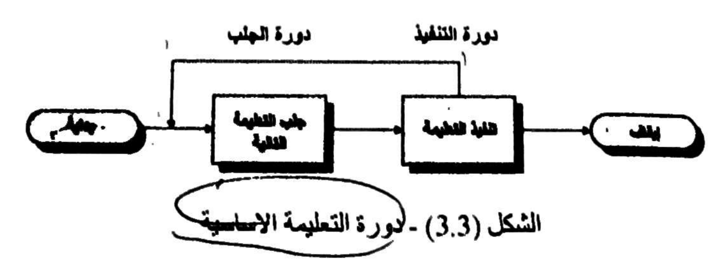
الشكل (3.2) - مكونات نظام الحاسب : **منظر عام**

3.2 وظيفة الحاسب

الوظيفة الأساسية التي يؤديها الحاسب هي تنفيذ البرنامج الذي يتألف من مجموعة من التعليمات المخزنة في الذاكرة ، فالمعالج هو الذي يقوم بالعمل الفعلي من خلال تنفيذ التعليمات المحددة في البرنامج .

يقدم هذا القسم لمحة عامة عن الخطوات الرئيسية لتنفيذ البرنامج في أبسط صورة حيث تتم معالجة التعليمة في خطوتين : المعالج يقرأ (يجلب) التعليمات من الذاكرة واحدة تلو الأخرى ثم ينفذ كل تعليمة ، ويتم تنفيذ البرنامج من خلال تكرار عملية جلب التعليمة وتنفيذها ، وقد ينطوي تنفيذ تعليمة ما على العديد من العمليات أو الاجراءات التي تعتمد على طبيعة التعليمة.

تسمى المعالجة اللازمة لاجراء تعليمة واحدة دورة التعليمة وذلك باستخدام الوصف المبسط المقدم سابقا والمكون من خطوتين ، والشكل (3.3) يصف دورة تعليمة ويشار إلى الخطوتين باسم دورة **الجلب** ، ودورة **التنفيذ** . ويتوقف تنفيذ البرنامج فقط إذا تم إيقاف تشغيل الجهاز أو حدوث خطأ ما ، أو تنفيذ تعليمة برنامج توقف عمل الحاسب.



الشكل (3.3) - دورة التعليمة الاساسية

3.3 جلب وتنفيذ التعليمة

في بداية دورة تعليمة يجلب المعالج التعليمة من الذاكرة . ففي المعالج النموذجي مسجل عداد البرنامج (PC) يحمل عنوان التعليمة التى ستجلب تاليا مالم يحدث خلاف ذلك ، ويقوم المعالج دائما بزيادة عداد البرنامج (PC) بعد جلب كل تعليمة بحيث يتم جلب التعليمة التالية في التسلسل (بمعنى أن التعليمة التالية هي الموجودة في أعلى عنوان بالذاكرة) .

فعلى سبيل المثال ، بأفتراض أن حاسباً له تعليمات تحتل كل واحدة منها كلمة واحدة من 16 خانة ثنائية (Bit) فى الذاكرة ، وأن عداد البرنامج يشير إلى الموقع 300 ، فالمعالج سوف يجلب التعليمة التالية والتى في الموقع 300 . وفى دورات التعليمة التالية سيتم جلب التعليمات من المواقع 301 ، 302 ، 303 ، وهكذا ، و يجوز تغيير هذا التسلسل كما سيوضح لاحقا .

يتم تحميل التعليمة التى تم جلبها في مسجل بالمعالج يعرف باسم مسجل التعليمة (IR) . والتعليمة تحتوي على ثنائيات تحدد الفعل الذى سيقوم به المعالج حيث يُفسر المعالج التعليمة وينفذ الإجراء (العمل) المطلوب ، وبشكل عام تصنف الأعمال إلى أربع فئات :

- المعالج - الذاكرة : عبارة عن نقل البيانات من المعالج إلى الذاكرة ، أو من الذاكرة إلى المعالج .
- المعالج - الإدخال/الإخراج : عبارة عن نقل البيانات من أو إلى جهاز طرفي عن طريق الانتقال بين المعالج و وحدة الإدخال/الإخراج .
- معالجة البيانات : يقوم المعالج بتنفيذ بعض العمليات الحسابية أو المنطقية على البيانات .

- التحكم : بناء على تنفيذ تعليمة ما قد يتغير تسلسل تنفيذ التعليمات .

فعلى سبيل المثال ، يجلب المعالج تعليمة من الموقع 149، الذى ينص على أن التعليمة التالية تكون من الموقع 182 . فالمعالج سوف يتذكر هذه الحقيقة من خلال وضع 182 فى عداد البرنامج ، ولذلك فى دورة الجلب التالية سيتم جلب التعليمة من الموقع 182 بدلا من 150 ، فتنفيذ التعليمة قد ينطوي على مزيج من مثل هذه الإجراءات .

بأخذ مثال بسيط وذلك بإستخدام معالج أفتراضي بالخصائص المدرجة في الشكل (3.4) حيث يحتوي المعالج على مسجل بيانات واحد يدعى مجمع (AC) . التعليمات والبيانات هي بطول 16 خانة ، وبذلك يكون من الملائم تنظيم الذاكرة باستخدام كلمات ذات 16 خانة. تنسيق التعليمة يقدم 4 خانات لرمز العملية وبذلك يمكن ان يكون هناك ما يصل الى $2^4 = 16$ رمز لعمليات مختلفة ، وكذلك قد يصل عدد كلمات الذاكرة التي يمكن معالجتها مباشرة الى $2^{12} = 4096$ (4K) كلمة .

15 … 4	3 … 0
العنوان	رمز التعليمة

(أ) تنسيق التعليمة

15 … 1	0
المقدار	الاشارة

(ب) تنسيق العدد الصحيح

عداد البرنامج (PC) : عنوان التعليمة
مسجل التعليمة (IR) : التعليمة الجارى تنفيذها
المجمع (AC) : تخزين مؤقت

(ج) مسجلات وحدة المعالجة المركزية

0001 = حمل المجمع من الذاكرة
0010 = خزن محتوى المجمع فى الذاكرة
0101 = اضف الى المجمع قيمة من الذاكرة

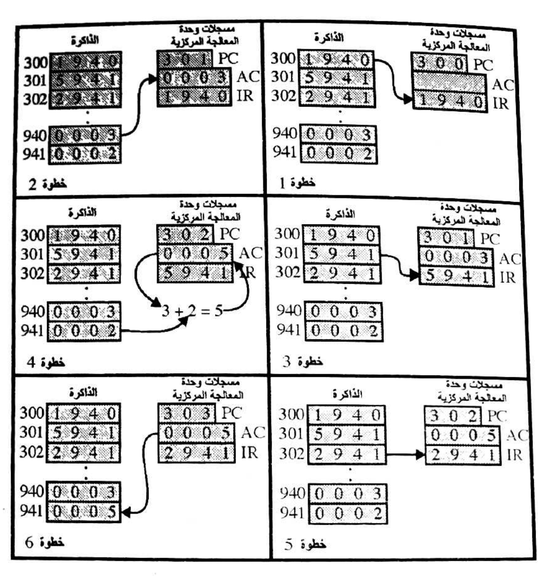
(د) قائمة جزئية من رموز تعليمات المعالج

الشكل (3.4) - خصائص المعالج الأفتراضي وتعليماته

... الشكل (3.5) جزئية من تنفيذ برنامج وكذلك الأجزاء ذات الصلة من ...ذاكرة ومسجلات المعالج . وتقوم الجزئية الموضحة من البرنامج بإضافة محتويات كلمة ذاكرة في العنوان (940) لمحتويات كلمة ذاكرة في عنوان (941) ، وتخزن النتيجة في العنوان الأخير ، ولأجراء ذلك ثلاث تعليمات مطلوبة ويمكن وصفها كثلاث دورات جلب وثلاث دورات تنفيذ :

1. عداد البرنامج (PC) يحتوي على القيمة (300) وهو عنوان التعليمة الأولى، يتم تحميل هذه التعليمة (القيمة [1940] بالحساب سادس عشري) في مسجل التعليمة (IR) ثم تتم زيادة عداد البرنامج (PC) واحد . لاحظ أن هذه العملية تنطوي على أستخدام مسجل عنوان ذاكرة (MAR) ، ومسجل الذاكرة المؤقت (MBR) كمسجلات وسيطة فى عملية النقل ، وللتبسيط سيتم تجاهل هذه المسجلات .
2. الخانات 4 الأولى (العدد السادس عشري الأول) في مسجل التعليمة (رمز التعليمة – كما هو مبين فى الشكل (3.4 – أ)) يشير إلى أنه سيتم تحميل بيانات الى مسجل المجمع (AC) وذلك بناءً على تفسير رمز التعليمة والمبين فى الشكل (3.4 – د)) ، وباقى 12 خانة (ثلاثة أعداد سادس عشرية) تُحدد عنوان البيانات التي سيتم تحميلها (940) .
3. يتم جلب التعليمة التالية [5941] من الموقع (301) ، ويزداد عداد البرنامج، لاحظ أن عداد البرنامج يتقدم خطوة بعد أتمام الجلب وقبل التنفيذ.
4. تتم إضافة المحتويات القديمة لمسجل المجمع (AC) مع محتويات الموقع (941) ، ويتم تخزين النتيجة في المجمع وذلك بناءً على تفسير رمز التعليمة (الجزء الأول من التعليمة – 4 خانات) والمبين فى الشكل (3.4 – د)) .

5. يتم جلب التعليمة التالية [2941] من الموقع (302) ، ويتقدم عداد البرنامج خطوة الى الأمام (يزداد واحد) .
6. تنفيذ التعليمة وذلك بتخزين محتويات المجمع في العنوان المشار إليه فى الجزء الثانى من التعليمة وهو الموقع (941) .



الشكل (3.5) – مثال على تنفيذ برنامج ما

(محتويات الذاكرة والمسجلات بالسادس عشرى)

تطلب هذا المثال ثلاث دورات تعليمة وتتألف كل منها من دورة جلب ، ودورة تنفيذ وذلك لإضافة محتويات الموقع (940) إلى محتويات (941) . ومع معالج

ذو طقم تعليمات أكثر تعقيدا سوف نحتاج إلى عدد أقل من الدورات . فبعض المعالجات القديمة على سبيل المثال تتضمن تعليمات تحتوي على أكثر من عنوان ذاكرة وبالتالي فإن دورة التنفيذ لهذه التعليمات الخاصة على مثل هذه المعالجات يمكن أن تنطوي على أكثر من تواصل واحد مع الذاكرة ، أيضا ، بدلا من التواصل مع الذاكرة فأن التعليمات قد تحدد عملية إدخال/إخراج .

على سبيل المثال ، معالج الحاسب (PDP-11) يتضمن تعليمة جمع حسابى يُعبر عنها رمزيا (ADD A,B) تقوم بجمع محتوى موقع الذاكرة (B) مع محتوى الموقع (A) ، وتخزين الناتج فى الموقع (A) ، وتحدث دورة التعليمة هذه حسب الخطوات التالية :

- جلب التعليمة(ADD A,B) .
- قراءة محتوي موقع الذاكرة (A) الى المعالج .
- قراءة محتوي موقع الذاكرة (B) الى المعالج . ومن أجل عدم ضياع محتوي (A) يجب ان يكون بالمعالج مسجلين أثنين على الأقل لتخزين قيم الذاكرة بدلا من مجمع واحد .
- جمع القيمتين .
- كتابة النتيجة من المعالج إلى موقع الذاكرة (A).

وهكذا، فإن دورة تنفيذ تعليمة خاصة قد تنطوي على أكثر من إشارة واحدة إلى الذاكرة . أيضا ، فبدلا من الأشارة للذاكرة قد تحدد التعليمة عملية إدخال/إخراج. ومع أخذ هذه الأعتبارات الإضافية يقدم الشكل (3.6) نظرة أكثر تفصيلا لدورة التعليمة الأساسية من الشكل (3.3) وهو على شكل رسم تخطيطي لحالات

التعليمة . ولأي دورة تعليمة معينة قد تلغى بعض الحالات والبعض الأخر قد يزار لأكثر من مرة ، ويمكن وصف الحالات على النحو التالي :

- حساب عنوان التعليمة : (IAC) تحديد عنوان التعليمة التالية اللازم تنفيذها . وعادة ماينطوى هذا على إضافة عدد ثابت إلى عنوان التعليمة السابقة . على سبيل المثال ، إذا كان طول كل تعليمة هو 16 خانة ثنائية والذاكرة منظمة بشكل كلمات بطول 16 خانة ثنائية ، فإنه تتم إضافة 1 إلى العنوان السابق (قفز كلمة بطول 16 خانة) ، خلاف ذلك ، فإذا كانت الذاكرة منظمة بشكل أن عنوان كل موقع يشير لكلمة بطول 8 خانة ثنائية ، ولنفس التعليمة (16-خانة) فإنه ستتم إضافة 2 إلى عنوان التعليمة السابقة (قفز كلمتين نتيجة أن التعليمة بطول كلمتين) .
- جلب التعليمة : (IF) قراءة التعليمة من موقع بالذاكرة إلى المعالج .
- عملية فك تشفير (تفسير) التعليمة : (IOD) تحليل أو تفسير التعليمة لتحديد نوع العملية التي سيتم تنفيذها والمعامل (المعاملات) الذى سيتم إستخدامه .
- حساب عنوان المعامل : (OAC) إذا كانت العملية تشتمل الإشارة إلى معامل في الذاكرة أو متاح عبر وحدات الإدخال/الإخراج ، يتم تحديد عنوان المعامل المطلوب .
- جلب المعامل : (OF) جلب المعامل من الذاكرة أو قراءته من وحدة الإدخال/الإخراج .
- العمل على (تشغيل) البيانات : (DO) تنفيذ العملية المشار إليها في التعليمة .
- تخزين المعامل : (OS) كتابة الناتج من تنفيذ التعليمة فى الذاكرة أو إلى وحدة الإدخال/الإخراج.

الشكل (3.6) – مخطط للدورة الأساسية لحالات التعليمة

تنطوى الحالات التي في الجزء العلوي من الشكل (3.6) على تبادل أو تواصل بين المعالج والذاكرة أو وحدة الإدخال/الإخراج . والحالات التى في الجزء السفلي من الرسم التخطيطي تتضمن حالات عمل داخل المعالج . فحالة حساب عنوان المعامل (OAC) تظهر مرتين وذلك لأنه قد تنطوى التعليمة على قراءة أو كتابة أو كليهما . ومع ذلك ، فإن العمل المنجز خلال تلك الحالة هو في الأساس نفسه في كلتا الحالتين ، ونحتاج فقط الى مميز للحالة (كتابة أم قراءة) . نلاحظ أيضا أن المخطط يسمح بمعاملات متعددة ونتائج متعددة ، وذلك لأن بعض التعليمات فى بعض الانظمة تتطلب ذلك . فعلى سبيل المثال ناتج تعليمة نظام الحاسب (PDP-11) الخاصة بالجمع (ADD A,B) يجهز بعد التسلسل التالي من الحالات :

(OS) ←(OAC) ←(DO) ←(OF) ←(OAC) ←(OF) ←(OAC) ←(IOD) ←(IF) ←(IAC)

أخيراً ، ويمكن فى بعض الأنظمة الحاسوبية أن تُحدد تعليمة واحدة العملية التي سيتم تنفيذها على شعاع (مصفوفة أحادية الأبعاد) من الأعداد أو الأحرف (سلسلة أحادية من الأحرف) أو البيانات كما هو مبين فى الشكل (3.6) وهذا ينطوي على تكرار جلب المعامل و/أو عمليات التخزين .

3.4 المقاطعات

تقريباً جميع الحاسبات الآلية توفر آلية تُمكن وحدات أخرى (الذاكرة أو وحدة الإدخال/الإخراج) من مقاطعة العمل العادى للمعالج . والجدول (3.1) يسرد الانواع الأكثر شيوعا من المقاطعات . وهنا نحن بحاجة إلى تقديم نبذة عن المقاطعات لأيضاح طبيعة دورة التعليمة فى هذه الحالة والآثار المترتبة لأستخدام المقاطعات على بنية الربط البيني للنظام . والقارئ لا يحتاج أن يركز في هذه المرحلة على تفاصيل انشاء ومعالجة المقاطعات ، ولكن التركيز هنا فقط على الأتصالات بين الوحدات التي تنتج هذه المقاطعات.

الجدول (3.1) – أنواع المقاطعات

النوع	سبب المقاطعة
برنامجي	ينشأ من حالة نتجت من تنفيذ تعليمة ما، مثل فيض حسابى ، تقسيم على صفر ، محاولة تنفيذ تعليمة غير مرخصة ، أو الأشارة الى موقع خارج المساحة المخصصة للمستخدم .
مُوقت	ينشأ من الساعة التى بالمعالج ، وهذا يسمح لنظام التشغيل بإجراء بعض الوظائف بانتظام .
الإدخال/الإخراج	ينشأ من متحكم الإدخال/الإخراج كنتيجة لنهاية عملية ما ، أو إشارة لحالة خطأ .
عطل مادى	ينشأ من عطل ما مثل فقدان التغذية الكهربية أو خطأ قراءة من الذاكرة .

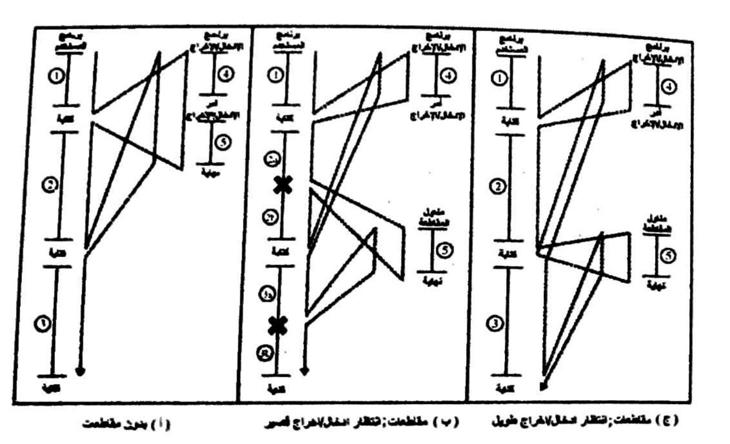
قدمت المقاطعات في المقام الأول باعتبارها وسيلة لتحسين كفاءة المعالجة . فعلى سبيل المثال معظم الأجهزة الخارجية هي أبطأ بكثير من المعالج ، ولنفترض أن معالج يرسل بيانات إلى طابعة باستخدام مخطط دورة التعليمة المبين في الشكل (3.3) ، فبعد كل عملية كتابة يجب على المعالج ان يتوقف وأن يبقى معطلاً حتى تستدرك الطابعة ، وطول هذه الوقفة قد يصل الى عدة مئات أو حتى آلاف من دورات التعليمات التى لا تنطوي على إشارة للذاكرة أو وحدة الإدخال/الإخراج أو التواصل معهما والتى يمكن تنفيذها خلال زمن التوقف (المعالج معطل) .

ومن الواضح أن هذا هدر كبير فى أستخدام المعالج والشكل (3.7 – أ) يوضح هذا الشأن . فبرنامج المستخدم ينفذ سلسلة من أستدعاءات الكتابة (WRITE) بينياً مع معالجة . والأجزاء البرمجية 1، 2، و3 تشير إلى سلسلة من التعليمات التي لا تنطوي على أدخال/إخراج . فإستدعاءات الكتابة (WRITE) هي لبرنامج جزئى للأدخال/الإخراج وهو أداة لإدارة موارد النظام والذى من شأنه تنفيذ عملية الإدخال/الإخراج فعلياً ، ويتألف برنامج الإدخال/الإخراج من ثلاثة أقسام :

- سلسلة متتابعة من التعليمات (الجزء رقم (4) في الشكل) للتحضير لبدء التشغيل الفعلي لعملية الإدخال/الإخراج . ويمكن أن يشمل هذا نسخ البيانات المخرجة في مخزن مؤقت خاص وإعداد المتغيرات لإدارة الجهاز الطرفى .
- أمر الإدخال/الإخراج الفعلي . ففي حالة عدم إستخدام المقاطعات فأن مجرد صدور هذا الأمر يعني انه على البرنامج الأنتظار حتى يقوم جهاز الإدخال/الإخراج بأداء الوظيفة المطلوبة ، وإنتظار البرنامج هو ببساطة عبارة عن إجراء عملية إختبار لتحديد ما إذا انتهت عملية الإدخال/الإخراج أم لا .

- سلسلة متتابعة من التعليمات (الجزء رقم (5) في الشكل) لإنهاء العملية ، ويمكن أن تشمل هذه وضع علامة تشير لنجاح أو فشل العملية .

لأن عملية الإدخال/الإخراج قد تستغرق وقتا طويلا نسبيا للأنتهاء لذلك فبرنامج الإدخال/الإخراج يبقي في انتظار عملية الإنتهاء ، وبالتالي برنامج المستخدم يتوقف في نقطة استدعاء الكتابة (WRITE) لفترة معتبرة من الزمن.



(أ) بدون مقاطعات (ب) مقاطعات؛ انتظار ادخال/اخراج قصير (ج) مقاطعات؛ انتظار ادخال/اخراج طويل

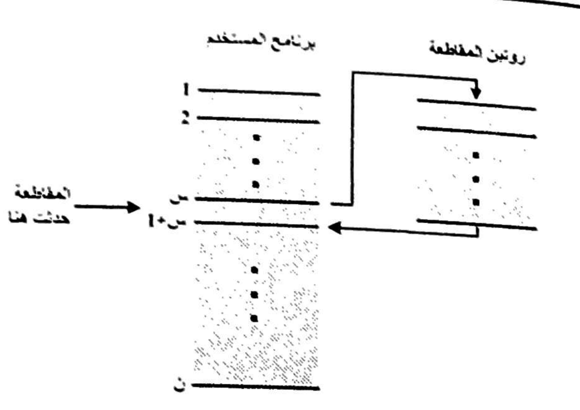
الشكل (3.7) - تسلسل التحكم فى برنامج بالمقاطعات و بدونها

3.4.1 المقاطعات ودورة التعليمة

فى دورة التعليمة بالمقاطعة يمكن للمعالج أن ينفذ تعليمات أخرى أثناء إجراء عملية الإدخال/الإخراج . وبالنظر الى تسلسل مسار التحكم المبين في الشكل

(3.7 – ب) ، وكما سبق ذكره فإن برنامج المستخدم يصل إلى نقطة لاستدعاء خدمة نظام في شكل استدعاء كتابة . وبرنامج الإدخال/الإخراج الذي يتم استدعاؤه في هذه الحالة يتكون فقط من تعليمات تحضيرية وأمر الإدخال/الإخراج الفعلي . و بعد أن يتم تنفيذ هذه التعليمات القليلة يعود التحكم من جديد إلى برنامج المستخدم ، وفي الوقت نفسه يكون الجهاز الخارجي مشغول في قبول البيانات من ذاكرة الحاسب وطباعتها ، وتتم عملية الإدخال/الإخراج هذه بالتزامن مع تنفيذ تعليمات في برنامج المستخدم . وعندما يصبح الجهاز الخارجي مستعداً لتقديم الخدمة بمعنى عندما يكون مستعداً لقبول مزيداً من البيانات من المعالج ، فعندها وحدة الإدخال/الإخراج الخاصة بهذا الجهاز الخارجي ترسل إشارة طلب مقاطعة إلى المعالج فيستجيب المعالج لها بتجميد عمل البرنامج الحالي ويتفرع الى برنامج خدمة خاص بذلك الجهاز ويعرف باسم **روتين المقاطعة** ، ويستأنف المعالج عمله الأصلي بعد ان يخدم الجهاز (انتهاء تنفيذ روتين المقاطعة) يشار إلى النقاط التي تحدث بها هذه المقاطعات بعلامة **(X)** في الشكل (3.7 – ب).

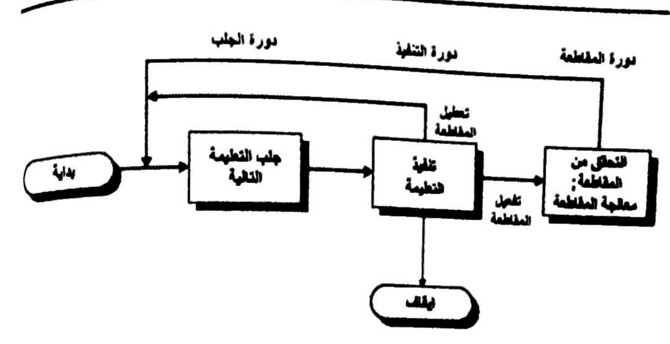
من وجهة نظر البرنامج المستخدم المقاطعة ليست سوى أنقطاع فى التسلسل الطبيعي للتنفيذ ، فعندما يتم الانتهاء من معالجة المقاطعة (روتين المقاطعة) يستأنف التنفيذ (الشكل- 3.8) . وبالتالي ، برنامج المستخدم لا يحتوى على أي تعليمات خاصة لخدمة المقاطعات ؛ المعالج ونظام التشغيل هما المسئولان عن وقف البرنامج المستخدم فى النقطة التى حدثت فيها المقاطعة والقفز لاجراء روتين المقاطعة ومن ثم أستئناف العمل به من نفس النقطة (النقطة التى أوقف فيها تنفيذ البرنامج) .



الشكل (3.8) - أنتقال التحكم بواسطة المقاطعة

ولأستيعاب المقاطعات تتم إضافة حالة المقاطعة إلى دورة التعليمة كما هو مبين في الشكل (3.9) . ففي حالة المقاطعة يتحقق المعالج لمعرفة ما إذا كانت قد حدثت أي مقاطعة مستدلاً على ذلك بوجود إشارة المقاطعة ، وإذا لم يكن هناك مقاطعات في الأنتظار عندها يتقدم المعالج الى دورة الجلب ويجلب التعليمة التالية فى البرنامج الحالي، وإذا كان هناك مقاطعة في الأنتظار فإن المعالج يقوم بمايلى :

- يُجمد تنفيذ البرنامج الحالي والجارى تنفيذه ويحفظ سياقه ، وهذا يعني حفظ عنوان التعليمة التالية للتنفيذ (المحتويات الحالية لعداد البرنامج) وغيرها من البيانات ذات الصلة بالنشاط الحالي للمعالج (محتويات المسجلات المختلفة داخل وحدة المعالجة المركزية).
- يُحمل عداد البرنامج بعنوان بداية البرنامج الجزئى (الروتين) المختص بمعالجة المقاطعة (عنوان أول تعليمة فى هذا الروتين) .



الشكل (3.9) – دورة التعليمة بالمقاطعة

وينتقل الأن المعالج إلى دورة الجلب ويجلب التعليمة الأولى في برنامج معالجة المقاطعة والذى سوف يخدم المقاطعة . وعموما يعتبر برنامج معالجة المقاطعة جزء من نظام التشغيل ، وعادة مايحدد هذا البرنامج طبيعة المقاطعة وينفذ كل الإجراءات المطلوبة . وفي المثال السابق ، يحدد برنامج معالجة المقاطعة أى من وحدات الإدخال/الإخراج أنتج المقاطعة وربما يتفرع إلى برنامج يزود تلك الوحدة ببعض البيانات ، وعند الأنتهاء من روتين معالجة المقاطعة يمكن للمعالج إستئناف تنفيذ برنامج المستخدم من نقطة التوقيف (الأنقطاع) .

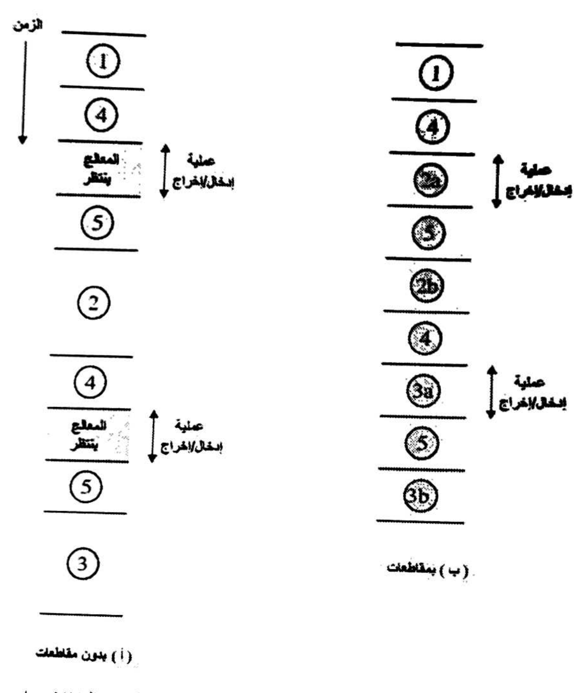
ومن الواضح أن هناك بعض الاضافات الى هذه العملية إذ يجب تنفيذ تعليمات إضافية (في روتين معالجة المقاطعة) لتحديد طبيعة المقاطعة والبت في الإجراء المناسب ، ومع ذلك ، فإنه نظرا للزمن الكبير نسبيا الذى سيضيع في الأنتظار أثناء عملية الإدخال/الإخراج التقليدية فإن المعالج يمكن إستخدامه بكفاءة أكبر بكثير بأستخدام المقاطعات .

ولتوضيح الزيادة في الكفاءة لاحظ الشكل (3.10) وهو عبارة عن رسم تخطيطي زمني أستناداً إلى أنتقال عناصر التحكم فى البرنامج المبين في الشكل (3.7 - أ) والشكل (3.7 - ب) .

فالشكل (3.7 - ب) والشكل (3.10) يفترضان أن الزمن اللازم لإنجاز عملية الإدخال/الإخراج قصير نسبياً : أي أقل من الزمن اللازم لإستكمال تنفيذ تعليمات ما بين عمليات الكتابة في برنامج المستخدم ، بمعنى انه قد تُطلب عملية كتابة أخرى قبل أن تستكمل أو تنجز عملية كتابة سابقة (الأنتهاء يتم بمقاطعة الإدخال/الإخراج للمعالج بعد أنجاز العملية أو أقرار حالة العملية) .

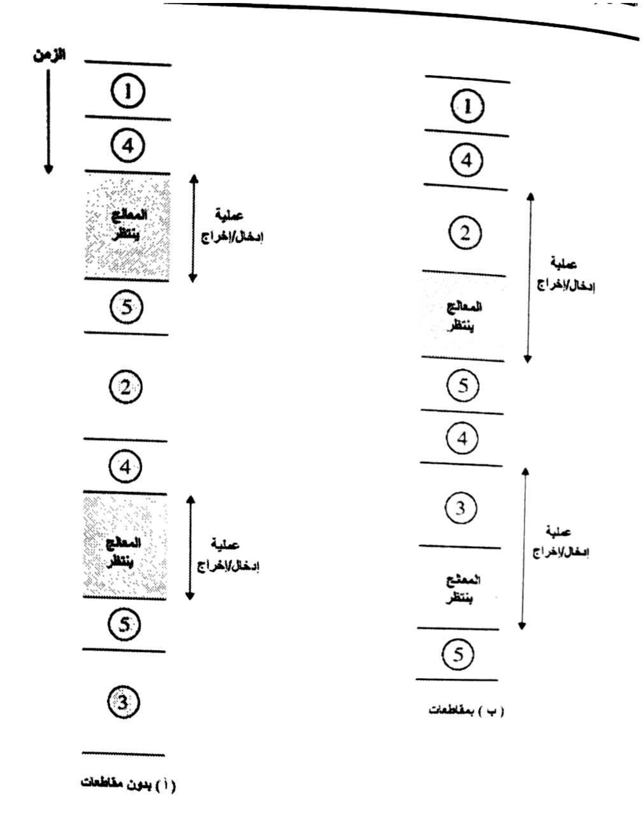
والحالة الأكثر نموذجية خاصة بالنسبة للأجهزة البطيئة مثل الطابعة هو أن عملية الإدخال/الإخراج تستغرق وقتا أطول بكثير من تنفيذ سلسلة من التعليمات الاخرى التى يتضمنها برنامج المستخدم .

يشير الشكل (3.7 - ج) الى هذا الوضع ، فتسلسل التحكم (التنفيذ) فى برنامج المستخدم يصل الاستدعاء الثانى للكتابة (WRITE) قبل أنتهاء عملية الإدخال/الإخراج الناتجة من الاستدعاء الأول وذلك ناتج لعدم انتهاء عملية الإدخال/الإخراج بسبب بطء أجهزة الإدخال/الإخراج ، والنتيجة أن برنامج المستخدم يمكث في تلك النقطة .



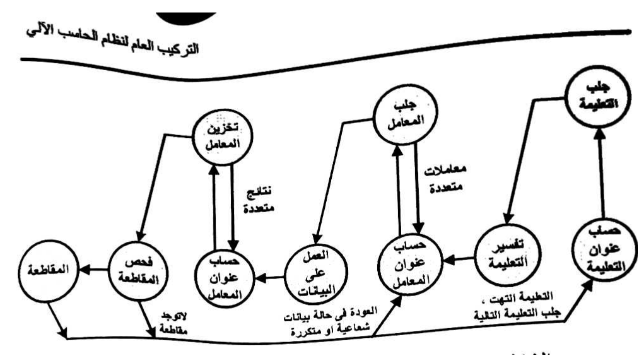
الشكل (3.10) – تزامن برنامج : أنتظار قصير للإدخال/الإخراج

وعند الأنتهاء من عملية الإدخال/الإخراج السابقة يمكن تنفيذ أستدعاء الكتابة الجديد وتبدأ عملية أدخال/إخراج جديدة . والشكل (3.11) يبين التزامن لهذا الوضع مع وبدون استخدام المقاطعة ، ويمكننا أن نرى أنه لا يزال هناك مكاسب في الكفاءة ؛ لأن جزءا من الزمن خلال عملية الإدخال/الإخراج يتداخل مع التنفيذ الجاري للتعليمات الأخرى التى يتضمنها برنامج المستخدم .



الشكل (3.11) - تزامن برنامج : أنتظار طويل للإدخال/الإخراج

ويبين الشكل (3.12) مخطط مُعدل لحالات دورة التعليمة الذى يشتمل على حالة معالجة المقاطعة .



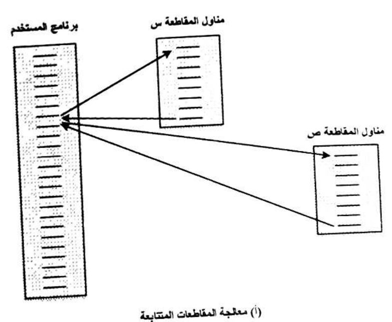
الشكل (3.12) - مخطط لدورة حالات التعليمة بالمقاطعة

3.4.2 المقاطعات المتعددة

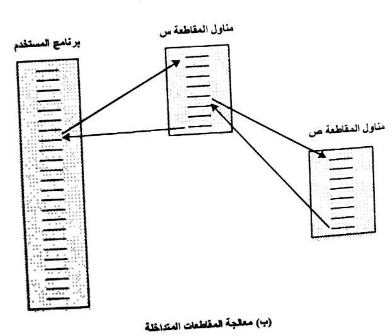
ركزت المناقشة السابقة فقط على وقوع مقاطعة واحدة ولنفترض الأن حدوث مقاطعات متعددة فى آن واحد بمعنى متزامنة فى الحدوث والتنفيذ ، وعلى سبيل المثال برنامج يستقبل بيانات على خط إتصالات ويطبع نتائج فى نفس الوقت. فالطابعة سوف تحدث مقاطعة كلما أنهت عملية طباعة ، ومتحكم خط الإتصال سيحدث مقاطعة كلما وصلت وحدة من البيانات ، وهذه الوحدة قد تكون إما حرفاً واحداً أو قالباً وهذا يتوقف على طبيعة ضبط الإتصال ، وعلى كل حال فإنه من الممكن أن تحدث أثناء معالجة مقاطعة الطابعة مقاطعة الإتصال .

نهجين يمكن اتخاذهما للتعامل مع المقاطعات المتعددة . الأول هو تعطيل المقاطعات أثناء معالجة مقاطعة . وتعطيل المقاطعة يعني ببساطة أن المعالج سيتجاهل أى إشارة طلب مقاطعة ، وإذا حدثت مقاطعة خلال هذا الوقت فأنها تبقى عادة مُجمدة وسيتم فحصها من قبل المعالج بعد تفعيل المعالج للمقاطعات من جديد . وهكذا ، فعندما يُنفذ برنامج المستخدم وتحدث مقاطعة ، فإنه يتم تعطيل المقاطعات فوراً . وبعد أنهاء روتين معالجة المقاطعة يتم تفعيل المقاطعات قبل

أستئناف برنامج المستخدم من تم يتحقق المعالج لمعرفة ما إذا حدثت أى مقاطعات إضافية لكى ينفذها حسب تسلسل وصولها . هذا النهج بسيط حيث يتم التعامل مع المقاطعات في ترتيب تسلسلي صارم (الشكل 3.13 – أ) .



(أ) معالجة المقاطعات المتتابعة



(ب) معالجة المقاطعات المتداخلة

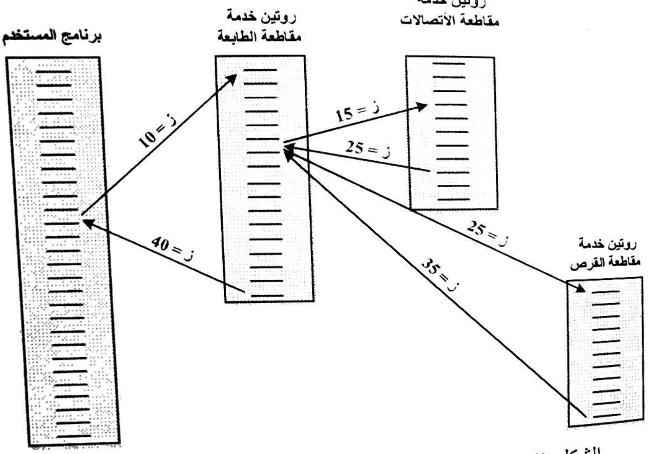
الشكل (3.13) – أنتقال التحكم فى المقاطعات المتعددة

العيب فى النهج السابق هو أنه لا يأخذ بعين الأعتبار الأولوية النسبية أو الأحتياجات في الوقت الحرج لبعض المقاطعات التى تحدث فى آن واحد لكن من

وحدات مختلفة الوظائف والاولويات . فعلى سبيل المثال عند وصول مدخلات **من خط الاتصال قد** يُحتاج لاستيعابها بسرعة لإفساح المجال لمزيد من **المدخلات**، وإذا لم تتم معالجة الدفعة الأولى من المدخلات قبل وصول الدفعة **الثانية قد** يتم فقدان بعض البيانات .

النهج الثاني هو تحديد أولويات المقاطعات والسماح للمقاطعة ذات الأسبقية **الأعلى** بإيقاف روتين معالجة مقاطعة ذات أسبقية أدنى (الشكل 3.13 – ب) . **كمثال** على النهج الثاني ، لاحظ المثال التالى : نظام من ثلاثة أجهزة **أدخال/إخراج :** طابعة ، قرص ، وخط الأتصالات ، مع أسبقيات متزايد من 2 ، **4 و** 5 على التوالي ، والشكل (3.14) يوضح التسلسل الممكن للأحداث . يبدأ برنامج المستخدم في ز= 0 ، عند ز= 10 تحدث مقاطعة الطابعة ؛ ويتم وضع **معلومات** المستخدم فى مكدس النظام وينتقل التنفيذ الى روتين خدمة مقاطعة الطابعة ، وفي اثناء تنفيذ هذا الروتين وعند ز= 15 تحدث مقاطعة الاتصالات ، **ولأن** خط الاتصالات لديه أسبقية أعلى من الطابعة يمنح حق المقاطعة ، وتتم مقاطعة روتين خدمة الطابعة وتوضع بيانات حالته فى المكدس ، وينتقل التنفيذ الى روتين خدمة مقاطعة الإتصالات . وأثناء تنفيذ روتين خدمة الاتصالات تحدث **مقاطعة** القرص عند ز= 20 ، ولأن هذه المقاطعة هى ذات اسبقية أقل تُجمد **ببساطة** ويستمر روتين الإتصالات فى التنفيذ حتى النهاية .
وعندما ينتهى روتين خدمة الإتصالات عند ز= 25 فإنه تتم أستعادة الحالة السابقة **للمعالج ، وهى** أستمرار تنفيذ روتين خدمة مقاطعة الطابعة . ومع ذلك ، فإنه قبل **أن يتم تنفيذ تعليمة واحدة من روتين مقاطعة الطابعة يمنح المعالج الأسبقية لمقاطعة القرص (أعلى اسبقية من الطابعة)** وينقل السيطرة إلى روتين خدمة

القرص . وفقط عند أنتهاء روتين خدمة القرص كاملاً (ز= 35) **يُستأنف تنفيذ** روتين خدمة الطابعة ، وعندما ينتهى هذا الروتين (ز= 40) **يرجع التحكم في** النهاية إلى برنامج المستخدم .



الشكل (3.14) – مثال على التسلسل الزمنى للمقاطعات المتعددة

3.5 وظيفة الإدخال/الإخراج

حتى الأن ناقشنا عمل الحاسب كنظام مُتحكم به من قبل المعالج ، **ولقد تم التركيز** فى المقام الأول على التفاعل بين المعالج والذاكرة ، **وقد ألمحت المناقشة للدور** المناط بأجهزة الإدخال/الإخراج ، **وسيناقش هذا الدور بالتفصيل في الفصل** (10)، **ولكن يمكن تناول ملخصاً موجزاً له هنا .**

وحدة الإدخال/الإخراج (على سبيل المثال ، مُتحكم القرص) يمكنها تبادل البيانات مباشرة مع المعالج . وكما أن المعالج يمكنه الشروع في القراءة أو الكتابة من الذاكرة وتحديد عنوان موقع معين ، ويمكن للمعالج أيضا قراءة البيانات أو كتابة البيانات من وإلى وحدة الإدخال/الإخراج ، وفى الحالة الأخيرة يحدد المعالج جهازاً معيناً مُتحكماً به من قبل وحدة أدخال/إخراج خاصة ، وبالتالي يحدث تسلسل تعليمات مماثل للذى فى الشكل (3.5) ولكن مع إستخدام تعليمات أدخال/إخراج بدلاً من تعليمات تؤشر للذاكرة .

وفي بعض الحالات ، من المستحسن أن يسمح لتبادل الإدخال/الإخراج أن يحدث مباشرة مع الذاكرة ، وفي مثل هذه الحالة يمنح المعالج وحدة الإدخال/الإخراج سلطة القراءة من أو الكتابة إلى الذاكرة بحيث ان النقل مابين الإدخال/الإخراج والذاكرة يحدث بدون الارتباط بالمعالج وخلال مثل هذا النقل فإن وحدة الإدخال/الإخراج تُصدر اوامر القراءة أو الكتابة إلى الذاكرة وتخفف من مسؤولية المعالج على التبادل ، وتعرف هذه العملية بالوصول المباشر للذاكرة (DMA) وسيتم شرحها فى الفصل (10) .

3.6 هياكل التوصيل البينى

يتكون نظام الحاسب من مجموعة مكونات أو وحدات من ثلاثة أنواع أساسية (المعالج والذاكرة و الإدخال/الإخراج) تتصل مع بعضها البعض . وفي الواقع ، فإن الحاسب عبارة عن شبكة من الوحدات الأساسية ، وبالتالي يجب أن يكون هناك مسارات لربط هذه الوحدات ، ويطلق على مجموعة المسارات التي تربط مختلف وحدات الحاسب بهيكل الربط البيني .

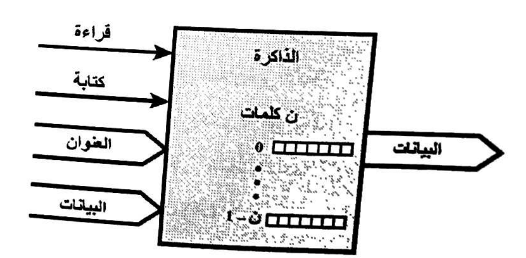
إن تصميم هذا الهيكل يعتمد على التبادل الذي يجب أن يتم بين وحدات نظام الحاسب . ويوضح الشكل (3.15) أنواع التبادلات التي يحتاجها كل نوع من الوحدات فى الحاسب وذلك بتوضيح المتطلبات الرئيسية من المدخلات والمخرجات لكل نوع من الوحدات :

- الذاكرة : عادة ما تتكون وحدة الذاكرة من عدد محدد (ن) من الكلمات المتساوية الطول . ولكل كلمة عنوان رقمي وحيد (0، 1، 2، ... ،ن-1) ، فكل كلمة من البيانات يمكن قراءتها من أو كتابتها إلى الذاكرة حسب ما تحدده إشارات تحكم القراءة والكتابة و عنوان موقع هذه الكلمة .
- وحدة الإدخال/الإخراج : بالنسبة لنظام الحاسب من الداخل الإدخال/الإخراج يشبه وظيفيا الذاكرة ، لذلك فوحدة الإدخال/الإخراج ترتبط بمسارات داخلية مع النظام . الوحدة يمكنها اجراء نوعين من العمليات هما القراءة والكتابة ، بالاضافة الى ذلك فإن وحدة الإدخال/الإخراج قد تتحكم فى أكثر من جهاز خارجي واحد ، ويمكن الاشارة إلى كل ارتباط مع جهاز خارجي بمنفذ و يُعطى عنواناً وحيداً (على سبيل المثال، 0، 1، ...، م-1) ، وهناك ايضاً مسارات خارجية مع الجهاز الخارجي لإدخال واخراج البيانات . وأخيرا ، قد تكون وحدة الإدخال/الإخراج قادرة على إرسال إشارات مقاطعة إلى المعالج .
- المعالج : يقرأ المعالج (تدخل) التعليمات والبيانات ، ويكتب (تخرج) البيانات بعد المعالجة ، ويستخدم إشارات تحكم للسيطرة على كامل العمل العام لنظام الحاسب ، وكذلك يستقبل إشارات المقاطعة .

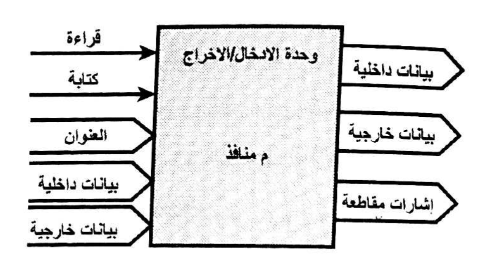
القائمة السابقة تحدد البيانات التي يتم تبادلها . فهيكل الربط البيني يجب أن يدعم الأنواع التالية من التحويلات :

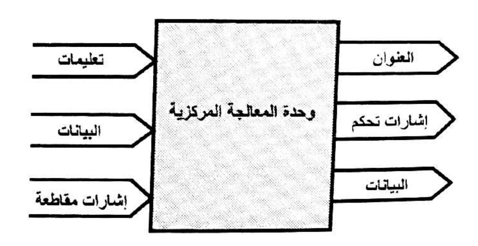
- **الذاكرة إلى المعالج** : المعالج يقرأ تعليمات أو وحدة بيانات من الذاكرة .
- **المعالج إلى الذاكرة** : المعالج يكتب وحدة بيانات إلى الذاكرة .
- **الإدخال/الإخراج إلى المعالج** : المعالج يقرأ بيانات من جهاز **الإدخال/الإخراج عن طريق** وحدة الإدخال/الإخراج .
- **المعالج إلى الإدخال/الإخراج** : المعالج يرسل البيانات إلى جهاز **الإدخال/الإخراج** .
- **الإدخال/الإخراج من أو إلى الذاكرة** : فى هاتين الحالتين ، يتم السماح لوحدة **الإدخال/الإخراج بتبادل البيانات** مباشرة مع الذاكرة ، ومن دون المرور عبر **المعالج ، وذلك بإستخدام** الوصول المباشر للذاكرة (DMA) .

على مر السنين ، تمت محاولة عدة هياكل للربط البينى ، والأكثر شيوعا هو **الناقل العام والناقل المتعدد** ، ويخصص ما تبقى من هذا الفصل لتقييم بنية الناقل.



الشكل (3.15 – أ) – تبادلات وحدات الحاسب





الشكل (3.15 – ب) – تبادلات وحدات الحاسب

3.7 ناقل الربط البيني

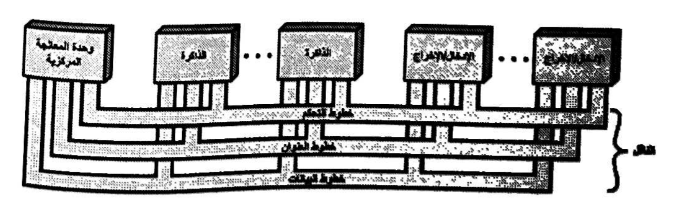
الناقل هو مسار الأتصال الذى يربط جهازين أو أكثر . وإحدى السمات المميزة للناقل هو أنه وسيلة نقل مشتركة حيث أن أجهزة متعددة ترتبط بالناقل ، والإشارة المرسلة من قبل أي جهاز متاحة للأستقبال من قبل **جميع الأجهزة الأخرى** المرتبطة بالناقل ، فإذا قام جهازان بالإرسال خلال **نفس الفترة الزمنية فأن** إشاراتهما تتداخل وتصبح مشوهة وبالتالي - **في وقت ما - جهاز واحد فقط يمكنه الأرسال بنجاح** .

نموذجيا ، يتضمن الناقل مسارات أو خطوط إتصال متعددة ، وكل خط له القدرة **على نقل إشارات متمثلة فى الثنائى 1** والثنائى 0 . زمنيا يمكن نقل سلسلة من **الخانات الثنائية عبر خط واحد** وبالتالى ناقل بمجموعة خطوط يمكن أستخدامه **لنقل مجموعة خانات بالتوازي** ، وعلى سبيل المثال ، وحدة بيانات من 8-خانات **يمكن نقلها على ناقل من 8 خطوط** .

يتضمن نظام الحاسب عدداً من النواقل المختلفة التى توفر مسارات مابين **المكونات** فى المستويات المتعددة من هيكلية نظام الحاسب . والناقل الذى يربط **المكونات الاساسية** (المعالج ، الذاكرة ، الإدخال/الإخراج) يدعى ناقل النظام ، **وهيكلية الربط البينى** تعتمد فى الشائع على استخدام ناقل نظام واحد او أكثر .

3.7.1 بنية الناقل

ناقل النظام يحتوى من 50 الى عدة مئات من الخطوط المنفصلة ، وكل خط **مخصص لوظيفة** او معنى محدد . ورغم وجود عدة تصاميم للناقل إلا أنه فى إى **ناقل يمكن تصنيف** الخطوط الى ثلاث مجموعات وظيفية (الشكل 3.16) : **خطوط بيانات وخطوط** عناوين وخطوط تحكم ، وبالإضافة لذلك ربما توجد **خطوط توزيع الطاقة** بحيث توفر التغذية الكهربية للوحدات المرفقة .



الشكل (3.16) – مخطط ربط الناقل

توفر خطوط البيانات المسار لأنتقال البيانات **مابين وحدات النظام ، وهذه** الخطوط - مع بعض - تدعى ناقل البيانات . **ناقل البيانات قد يتضمن 32 أو 64** أو 128 خط منفصل أو ربما أكثر ، وعدد **الخطوط يشار إليه بعرض ناقل** البيانات. ونتيجة لأن كل خط يحمل فقط **خانة واحدة فى كل مرة ، فإن عدد** الخطوط تحدد عدد الخانات التى يمكن نقلها **بالتوازى فى كل مرة . فعرض ناقل** البيانات هو مقياس أساسى لتحديد الكفاءة العامة للنظام . **على سبيل المثال ، إذا** كان عرض ناقل البيانات 32 خانة والتعليمة **طولها 64 خانة ، ففى هذه الحالة** يجب على المعالج التواصل مع الذاكرة **مرتين خلال كل دورة تعليمة** .

تستخدم خطوط العناوين للدلالة على مصدر أو **وجهة البيانات التى على ناقل** البيانات . على سبيل المثال ، اذا رغب المعالج فى **قراءة كلمة (8 ، 16 ، أو 32** خانة) من البيانات من الذاكرة فإنه يضع **عنوان الكلمة المرغوب فيها على خطوط** العناوين . ومن الواضح أن عرض ناقل **العناوين يحدد الحد الأقصى للسعة** المحتملة لذاكرة النظام . وعلاوة على ذلك ، **خطوط العناوين يمكن أن تستخدم** لتحديد منافذ الإدخال/الإخراج . ونموذجيا **الخانات العليا في الترتيب تستخدم** لأختيار وحدة معينة على الناقل ، بينما تحدد **الخانات الدنيا فى الترتيب موقع** ذاكرة أو منفذ إدخال/إخراج فى تلك الوحدة . **فعلى سبيل المثال ، ناقل عناوين** مكون من 8-خانات ، فالعنوان 01111111 **وأدناه يمكن أن تؤشر لمواقع فى** وحدة ذاكرة (الوحدة - 0) بسعة **128 كلمة للوحدة ، والعنوان 10000000 وما** فوقه تؤشر لأجهزة مرتبطة بوحدة **الإدخال/الإخراج (الوحدة - 1)** .

تستخدم خطوط التحكم **للسيطرة على الوصول وإستخدام خطوط البيانات** والعناوين وذلك لأن **خطوط البيانات والعناوين مُشتَركة مابين جميع مكونات**

النظام ، ويجب أن يكون هناك وسيلة ما للسيطرة على استخدامها . فإشارات التحكم ترسل معلومات السيطرة والتزامن ما بين وحدات النظام ، وإشارات التزامن تشير إلى صحة توقيت البيانات والعناوين ، أما إشارات السيطرة فإنها تحدد العمليات الواجب أداؤها . نموذجيا ، تشمل خطوط التحكم :

- **كتابة فى الذاكرة :** يسبب كتابة البيانات التى على متن الناقل في الموقع المعنون .
- **قراءة من الذاكرة :** يسبب قراءة البيانات من الموقع المعنون و وضعها على متن الناقل .
- كتابة فى وحدة الإدخال/الإخراج : يسبب أخراج البيانات التى على متن الناقل إلى منفذ أدخال/إخراج معنون.
- قراءة من وحدة الإدخال/الإخراج : يسبب إدخال البيانات من منفذ أدخال/إخراج معنون و وضعها على متن الناقل .
- أرسال التسليم : تشير إلى أن البيانات قد وُضعت على متن الناقل أو قبلت منه .
- طلب الناقل : يشير إلى أن وحدة ما تريد السيطرة على الناقل .
- منح الناقل : يشير إلى أن الوحدة الطالبة منحت السيطرة على الناقل .
- طلب المقاطعة : يشير إلى أن هناك مقاطعة لا تزال تنتظر .
- إقرار المقاطعة : أشعار بأن المقاطعة المنتظرة قد تم التعرف عليها.
- **الساعة :** تستخدم لمزامنة العمليات .
- **الإعادة :** تهيئة جميع الوحدات .

و يتم عمل الناقل على النحو التالي : إذا رغبت وحدة فى أرسال بيانات **الى وحدة** أخرى ، فيجب أن تفعل شيئين : (1) الحصول على حق استخدام الناقل ، (2) نقل البيانات على متن الناقل . وأذا أرادت وحدة طلب بيانات من وحدة أخرى ، فإنه يجب (1) الحصول على حق استخدام الناقل ، (2) نقل الطلب إلى وحدة أخرى من خلال خطوط التحكم والعنونة المناسبتين ويجب بعدها أنتظار الوحدة الثانية كى ترسل البيانات .

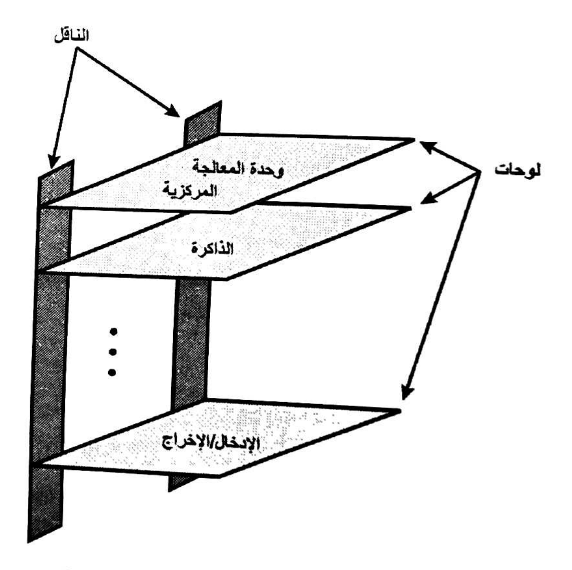
مادياً ، نظام الناقل هو في الواقع عدد من الموصلات الكهربائية المتوازية . **ففي** الناقل التقليدى هذه الموصلات هي خطوط معدنية محفورة في بطاقة أو لوحة (لوحة الدوائر المطبوعة) . والناقل يمتد عبر مكونات النظام كافة ، وكل واحدة منها ترتبط مع بعض أو كل خطوط الناقل و الشكل (3.17) يصور الترتيب الفعلي التقليدى .

وفي هذا الشكل يتألف الناقل من عمودين رأسيين من الموصلات ، وعلى طول الأعمدة وبمسافات منتظمة هناك نقاط تعليق على شكل فتحات تمتد أفقيا لدعم لوحة الدوائر المطبوعة . المكونات الرئيسية للنظام موضوعة على لوحة أو أكثر وتتعشق مع الناقل من خلال فتحات أفقية ويقع هذا الترتيب بالكامل داخل هيكل مغلف .

ولا يزال هذا المخطط يستخدم فى بعض الناقلات المرتبطة مع نظام الحاسب ، ومع ذلك تميل النظم الحديثة إلى وضع كافة المكونات الرئيسية على متن لوحة واحدة مع أكثر من عنصر على الشريحة الواحدة كما فى المعالج ، وبالتالي ، فإن الناقل الذى على الشريحة يربط المعالج والذاكرة السريعة ، وفي حين أن الناقل

***الذى على اللوحة* قد يربط المعالج مع الذاكرة الرئيسية والمكونات الأخرى (*هيكلية من الناقلات*) .**

وهذا الترتيب هو الأكثر ملائمة حيث يُمكن من الحصول على نظام حاسب صغير **قابل للتوسعة لاحقا** (مزيداً من الذاكرة وعدد أكثر من وحدات الإدخال/الإخراج) **وذلك بإضافة المزيد من اللوحات فإذا فشل مكون على لوحة ما يمكن بسهولة إزالة هذه اللوحة واستبدالها .**



الشكل (3.17) – التركيب النموذجى لمعمارية الناقل

3.7.2 هيكلية الناقلات المتعددة

إذا تم توصيل عدد كبير من الأجهزة إلى الناقل فأن الأداء سيسوء ، وذلك للأسباب الرئيسية التالية :

1. بشكل عام ، كلما زاد عدد الأجهزة المرفقة إلى الناقل ، زاد طول الناقل ، وبالتالي زاد تأخير التوصيل . ويُحدد هذا التأخير الزمن الذي تستغرقه الأجهزة لتنسيق أستخدام الناقل ، فعند تمرير السيطرة على الناقل من جهاز إلى آخر باستمرار يمكن لتأخيرات التوصيل من التأثير بشكل ملحوظ على الأداء.
2. قد يصبح الناقل عنق الزجاجة فى النظام عندما يقترب مجموع طلبات نقل البيانات الى قدرة الناقل . ويمكن مواجهة هذه المشكلة إلى حد ما عن طريق زيادة معدل البيانات التى يمكن للناقل أن يحملها وذلك باستخدام ناقلات أوسع (على سبيل المثال ، زيادة ناقل البيانات من 32-خانة الى 64-خانة) . ومع ذلك لأن معدل البيانات التي تولدها الأجهزة المتصلة بالناقل (على سبيل المثال ، متحكم الرسومات والفيديو ، وروابط شبكة الاتصال) تزداد بسرعة مما أدى بالناقل الى أن يخسر هذا السباق في نهاية المطاف .

وفقا لذلك ، فإن معظم أنظمة الحاسب تستخدم ناقلات متعددة موضوعة عادةً في تسلسل هرمي ، والشكل (3.18 - أ) يبين نموذج لهذه البنية التقليدية . **فالناقل المحلي يربط المعالج إلى الذاكرة السريعة والتي يمكن أن تدعم واحداً أو أكثر من الأجهزة المحلية** . ومُتحكم الذاكرة السريعة **يربط الذاكرة السريعة مع الناقل المحلي وناقل النظام والذي بدوره يربط كل وحدات الذاكرة الرئيسية .**

إن إستخدام بنية الذاكرة السريعة يعزل المعالج عن متطلبات التواصل باستمرار **مع الذاكرة الرئيسية** ، وبالتالي يتيح نقل الذاكرة الرئيسية من الناقل المحلي الى **ناقل النظام وبهذه الطريقة** فإن نشاط الإدخال/الإخراج من و إلى الذاكرة الرئيسية **عبر ناقل النظام لا يتداخل مع نشاط المعالج .**

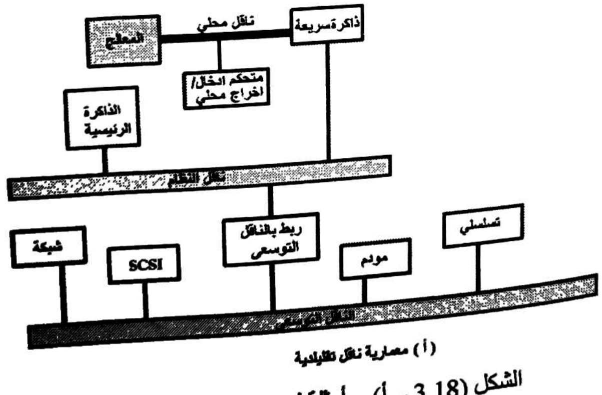
ومن الممكن ربط مُتحكمات الإدخال/الإخراج مباشرة على ناقل النظام ، ولزيادة **الكفاءة يمكن** الإستفادة من ناقل توسعى أو أكثر لهذا الغرض بحيث أن الناقل **التوسعى ينسق** نقل البيانات بين ناقل النظام و مُتحكمات الإدخال/الإخراج **المرتبطة بالناقل التوسعى** . وهذا الترتيب يسمح للنظام بدعم مجموعة واسعة من **أجهزة الإدخال/الإخراج** ، وفي الوقت نفسه يعزل حركة مرور الذاكرة إلى **المعالج عن حركة** مرور الإدخال/الإخراج.

يبين الشكل (3.18 – أ) بعض الأمثلة النموذجية لأجهزة الإدخال/الإخراج التي **يُمكن أن تُرفق** بالناقل التوسعى . فروابط الإتصالات الشبكية تشمل شبكة **الإتصالات المحلية** (LAN) مثل 10– ميغا ثُمانَ في الثانية إيثرنت (Ethernet) **وشبكة الاتصالات الموسعة** (WAN) مثل تقنية شبكة التبديل – بالحزمة **(Packet-Switching)** . ونظام الربط الحاسوبى (SCSI) هو نوع من تقنية **الناقلات التى** تستخدم لدعم محركات الأقراص المحلية وغيرها من الأجهزة **الطرفية ، والمنفذ التسلسلي** يمكن أن يستخدم لربط الطابعة أو الماسح الضوئي .

إن هذه العمارة التقليدية للناقل تتسم بالكفاءة إلى حد معقول ولكنها تنخفض كلما نظرنا الى الأداء المتزايد لأجهزة الإدخال/الإخراج . واستجابة لهذه المطالب **المتزايدة أتخذت الصناعة منهجاً مشتركاً فى بناء ناقلات عالية السرعة متكاملة بشكل وثيق مع بقية النظام ولا تتطلب سوى جسر بين ناقل المعالج والناقل العالي**

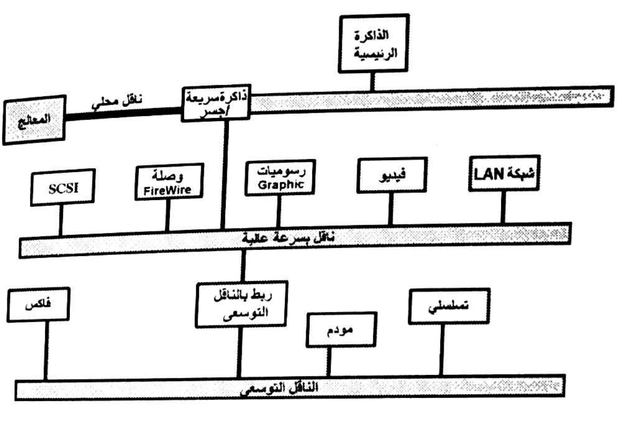
السرعة ، وهذا الترتيب يعرف أحيانا باسم **معمارية ميزانين** . **ويُظهر الشكل** (3.18 – ب) النمط النموذجى لهذا النهج . ومرة أخرى ، **هناك ناقل محلي يربط** المعالج إلى مُتحكم الذاكرة السريعة والذي بدوره متصل بناقل النظام الذى **يدعم** الذاكرة الرئيسية وقد تم دمج مُتحكم الذاكرة السريعة مع جسر أو وحدة تخزين مؤقت متصلة بناقل عالي السرعة .

ويدعم هذا الناقل العالي السرعة الربط بالشبكات المحلية عالية السرعة مثل الإيثرنت السريعة 100–ميغاخانة ثنائية في الثانية ، ومُتحكمات الفيديو ومحطات عمل الرسومات ، فضلاً عن الربط مع مُتحكم ناقل الطرفيات المحلية مثل (SCSI) و (FireWire) ، وهذا الأخير هو تنظيم لناقل عالي السرعة مصمم خصيصا لدعم أجهزة الإدخال/الإخراج ذات القدرات العالية . ولا تزال تعتمد ألأجهزة ذات السرعة المنخفضة على الناقل التوسعى مع رابط تخزين مؤقت لحركة المرور بين الناقل التوسعى والناقل العالي السرعة .



(أ) معمارية ناقل تقليدية

الشكل (3.18 – أ) – أمثلة لمعماريات الناقل



(ب) معمارية عالية الأداء

الشكل (3.18 – ب) – أمثلة لمعماريات الناقل

فميزة هذا الترتيب هى أن الناقل عالي السرعة يجعل الأجهزة المرتفعة الطلب **ملاصقة للمعالج** ومتكاملة معه وفي الوقت نفسه مستقلة عن المعالج ، وبالتالي **يمكن تحمل الاختلافات** مابين سرعة المعالج وسرعة الناقل عالي السرعة **وتعريف خطوط الإشارة** . والتغيير في معمارية المعالج لا تؤثر على الناقل عالي **السرعة ، والعكس بالعكس** .

3.7.3 عناصر تصميم الناقل

على الرغم من وجود مجموعة متنوعة من التصميمات المختلفة للناقلات فإن هناك عدداً قليلاً من المعالم الأساسية أو عناصر التصميم هي التي تُصنف وتُميز الناقلات . ويسرد الجدول (3.2) العناصر الرئيسية فى التصميم .

الجدول (3.2) - عناصر تصميم الناقل

النوع : ▪ مخصص ▪ مشترك	**عرض الناقل :** ▪ عنوان ▪ بيانات
طريقة التحكيم : ▪ مركزى ▪ موزع **التزامن :** ▪ متزامن ▪ غير متزامن	**شكل نقل البيانات :** ▪ قراءة ▪ كتابة ▪ قراءة – تعديل – كتابة ▪ قراءة – بعد – كتابة ▪ قالب

3.7.3.1 أنواع الناقل

يمكن تصنيف الناقل إلى نوعين عامين : **مخصص أو مُشتَرك . فأما خط الناقل** المُخصص فيتم تخصيصه بشكل دائم إما إلى وظيفة ما أو **لمجموعة فرعية من** مكونات الحاسب المادية . ومثال التخصيص الوظيفي هو أستخدام **خطين** منفصلين مخصصين إحدهما للعناوين و الثانى للبيانات وهو **أمر شائع في العديد** من الناقلات غير أن هذا ليس ضرورياً . وعلى سبيل المثال ، **يمكن أن تُنقل** معلومات العنوان والبيانات على نفس مجموعة الخطوط باستخدام **خط تحكم** بصلاحية العنوان . ففي بداية نقل البيانات **يتم وضع العنوان على متن الناقل ويتم** تنشيط خط صلاحية العنوان ، وعند هذه النقطة ، **كل وحدة (مرتبطة بالناقل)** لديها فترة محددة من الوقت لنسخ العنوان وتحديد ما إذا كانت هى **المقصودة ، ثم** تتم إزالة العنوان من على الناقل **وتستخدم خطوط الناقل نفسه لاحقا لنقل البيانات**

في القراءة أو الكتابة ، ويعرف هذا الأسلوب من استخدام نفس الخطوط لأغراض **متعددة بالمُشاركة الزمنية .**

إن ميزة أسلوب المشاركة الزمنية هو أستخدام عدد أقل من الخطوط مما يوفر **مساحة وكذلك التكلفة .** والعيب هو الحاجة لدوائر أكثر تعقيداً داخل كل وحدة **للتحكم في الناقل** (تمييز الموجود على الناقل بيانات أم عنوان) . أيضا ، هناك **أنخفاض محتمل** في الأداء بسبب أن أحداث معينة قد تشترك في نفس الخطوط لا **يمكن أن تحدث** بالتوازى .

ويشير التخصيص المادي إلى أستخدام ناقلات متعددة ، كل منها يَربط فقط **مجموعة فرعية** من الوحدات . والمثال النموذجي هو أستخدام ناقل أدخال/إخراج **لربط جميع** وحدات الإدخال/الإخراج ، ثم يتم توصيل هذا الناقل إلى الناقل **الرئيسي من خلال** نوع ما من وحدات تحويل الإدخال/الإخراج . والميزة المحتملة **للتخصيص المادي** هى الإنتاجية العالية نتيجة أن التزاحم أقل في الناقل والعيب **هو زيادة حجم** وتكلفة النظام .

3.7.3.2 طريقة التحكيم

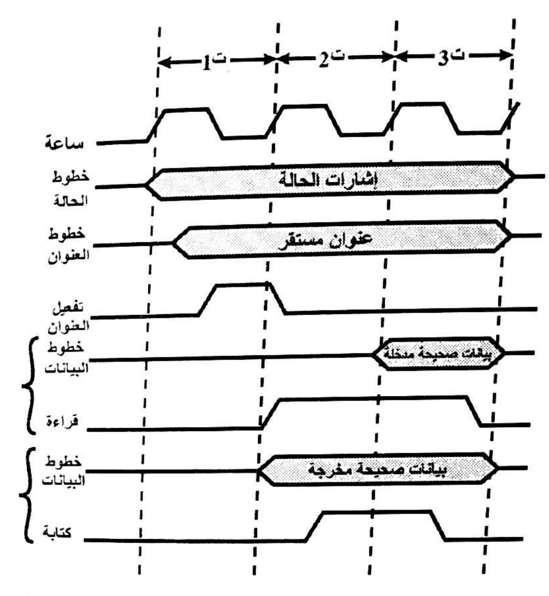
في جميع النظم البسيطة ، أكثر من وحدة قد تُريد السيطرة على الناقل ، فعلى **سبيل المثال ، قد تحتاج** وحدة أدخال/إخراج للقراءة أو الكتابة مباشرة إلى الذاكرة **دون إرسال البيانات إلى المعالج** ، ولأن وحدة واحدة فقط في وقت ما يمكن أن **تستخدم الناقل بنجاح** (نقل بيانات) بين مجموعة وحدات متصلة بالناقل كانت **هناك الحاجة لأسلوب تحكيم لأستغلال الناقل** . ويمكن تصنيف الأساليب المختلفة على **أنها إما مركزية أو موزعة . ففي المركزية** ، جهاز واحد ويشار إليه بوحدة

تحكم الناقل أو **المُحَكم** (Arbiter) وهو المسئول عن **تخصيص الحصة الزمنية** على الناقل (بين مجموعة وحدات) وقد يكون الجهاز وحدة **منفصلة أو جزء من** المعالج .

وفي **الموزعة** ، ليس هناك وحدة تحكم مركزية وبدلا من ذلك ، **كل وحدة تحتوي** على وحدة تحكم في الوصول الى الناقل وكل الوحدات **تعمل معا لتبادل أستخدام** الناقل . وفى ألاسلوبين للتحكيم فإن الغرض هو تحديد جهاز واحد ، **سواء المعالج** أو وحدة الإدخال/الإخراج ، كمُسيطر (السَيد) على الناقل . **ويجوز للمُسيطر** الشروع في نقل البيانات (على سبيل المثال ، القراءة أو الكتابة) **مع الأجهزة** الأخرى التي تقوم بدور المستخدم (الخادم) في هذا التبادل .

3.7.3.3 التزامن

يشير التزامن للطريقة التي يتم بها تنسيق الأحداث على متن الناقل . **يستخدم** الناقل إما التوقيت **المتزامن** أو التوقيت **غير المتزامن . ففى التوقيت المتزامن** (Synchronous) يتم تحديد وقوع الإحداث على متن الناقل **بواسطة نبضة** الساعة ، وفى هذه الحالة يتضمن الناقل **خطاً للساعة بحيث ترسل عليه تسلسلاً** تناوبياً منتظماً من 1 و 0 متكافئ المدة (نبضات) . **ويشار إلى انتقال واحد من 1** الى 0 كدورة ساعة أو **دورة الناقل** (Clock cycle or Bus cycle) **وهى تُحدد** الفسحة الزمنية للنبضة وكل الأجهزة الأخرى **على متن الناقل يمكنها قراءة خط** نبضة الساعة وجميع الأحداث **تبدأ عند بداية دورة نبضة الساعة ، ويبين الشكل** (3.19) نموذجاً مبسطاً **لمخطط التوقيت المتزامن لعمليات القراءة والكتابة .**

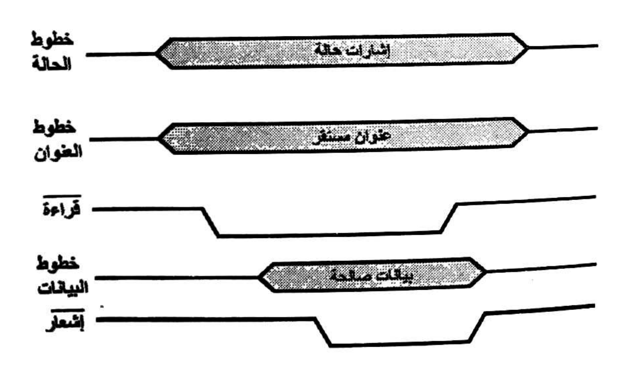


الشكل (3.19) – مخطط التوقيت المتزامن لعمليات الناقل

أما الإشارات الأخرى للناقل فقد تتغير عند الحافة الأمامية لإشارة نبضة الساعة **(مع تأخير طفيف لرد الفعل)** ، ومعظم الأحداث على الناقل تحتل دورة ساعة **واحدة . وفي هذا المثال البسيط** (الشكل 3.19) ، **المعالج يضع عنوان الذاكرة على خطوط العنونة خلال دورة الناقل الأولى وربما يؤكد ذلك بخطوط خاصة بالحالة . وعندما تستقر خطوط العنوان** ، فإن المعالج يُصدر إشارة تمكين العنوان. **فى عملية قراءة ، يُصدر المعالج أمر قراءة في بداية الدورة الثانية ، وعندها وحدة الذاكرة تتعرف على العنوان وبعد تأخير لدورة واحدة تضع البيانات على خطوط البيانات ومن تم يقرأ المعالج البيانات من خطوط البيانات ويُخفض إشارة**

فى عملية الكتابة ، يضع المعالج البيانات على خطوط البيانات **في بداية الدورة** الثانية ، ويُصدر أمر كتابة بعد أن تستقر خطوط البيانات و **وحدة الذاكرة تنسخ** المعلومات من خطوط البيانات خلال دورة الناقل الثالثة .

أما فى التوقيت غير المتزامن (Asynchronous) ، **فإن وقوع حدث ما على** الناقل يلي ويعتمد على وقوع حدث سابق . وفي مثال القراءة **البسيط المبين فى** الشكل (3.20 – أ) يضع المعالج العنوان وإشارات الحالة **على متن الناقل .**



الشكل (3.20 – أ) – **مخطط التوقيت غير المتزامن لعمليات الناقل**

خطوط الحالة — إشارات حالة
خطوط العنوان — عنوان مستقر
خطوط البيانات — بيانات صالحة
كتابة
اشعار

(ب) دورة كتابة ناقل النظام

الشكل (3.20 – ب) – مخطط التوقيت غير المتزامن لعمليات الناقل

وبعد أستقرار هذه الإشارات يَصدر أمر القراءة ، ويشير إلى وجود إشارات تحكم **و عنوان صالحين** . وتترجم وحدة مناسبة من الذاكرة العنوان وتستجيب عن **طريق وضع** البيانات على خط البيانات ، وعندما تستقر خطوط البيانات فإن **وحدة الذاكرة** تُنشط خط التسليم ليومئ إلى المعالج بأن البيانات متاحة . وعندما **تقرأ الوحدة المسيطرة** (على الناقل) البيانات من خطوط البيانات فإنها تنشط إشارة **القراءة وهذا يتسبب** فى خفض خطوط البيانات و التسليم لوحدة الذاكرة . وأخيراً، **عندما يتم خفض** خط التسليم فإن المسيطر يزيل معلومات العنوان .

ويبين الشكل (3.20 – ب) عملية كتابة غير متزامنة بسيطة . وفي **هذه الحالة يضع المسيطر البيانات** على خط البيانات في نفس الوقت الذي يضع فيه إشارات **من الحالة وخطوط العنوان** ، و تستجيب وحدة الذاكرة إلى أمر الكتابة عن طريق ..

نسخ البيانات من خطوط البيانات ومن **ثم التأكيد على خط التسليم ، ثم يخفض** المسيطر إشارة الكتابة و تخفض وحدة الذاكرة **إشارة التسليم** .

التوقيت المتزامن هو أبسط فى التنفيذ والأختبار . **ومع ذلك ، فانه أقل مرونة من** التوقيت غير المتزامن وذلك لأن جميع الأجهزة على **متن الناقل المتزامن مرتبطة** بمعدل ساعة ثابت (نبضة) مما **يجعل النظام لا يستفيد من جهاز عالى الأداء ،** ولكن مع التوقيت غير **المتزامن يمكن لخليط من الأجهزة البطيئة والسريعة** وباستخدام تقنية قديمة وجديدة أن تشترك في الناقل .

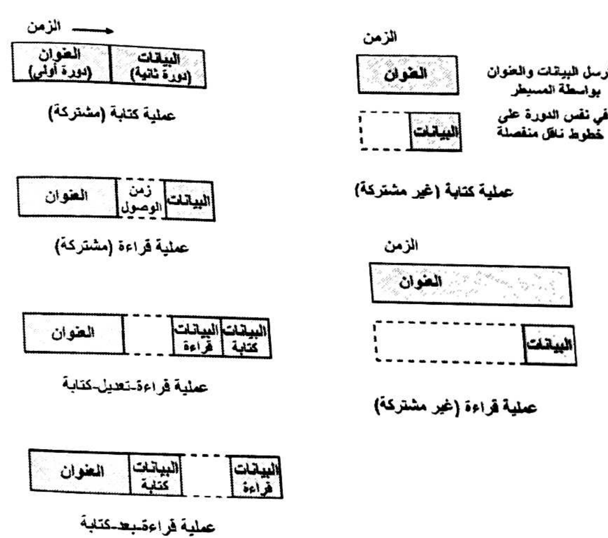
3.7.3.4 عرض الناقل

لقد تناولنا سابقا مفهوم عرض الناقل ، **فعرض ناقل البيانات له تأثير مباشر على** أداء النظام : فكلما اتسع نطاق ناقل البيانات زاد عدد **الخانات المنقولة في الزمن** الواحد (بالتوازى) . وعرض ناقل العنوان له تأثير على **قدرة النظام : ناقل عناوين** اعرض يُمكن من التأشير إلي مجموعة اكبر من **المواقع (التواصل معها).**

3.7.3.5 أنماط نقل البيانات

وأخيرا ، يمكن للناقل ان يدعم أنماط مختلفة من **نقل البيانات ، وكما هو موضح** في الشكل (3.21) . **فجميع الناقلات تدعم نقل الكتابة (من المُسيطر الى** المستخدم) والقراءة (من قبل المستخدم للمُسيطر) **. وفي حالة وجود ناقل مُشترك** العنوان/البيانات يتم استخدام **الناقل أولا لتحديد العنوان ثم لنقل البيانات . ولعملية** القراءة عادةً مايكون **هناك انتظار بينما يتم جلب البيانات من المستخدم لتوضع** على متن الناقل . **فى القراءة أو الكتابة قد يكون هناك تأخير أيضاً إذا كان من** الضرورى **أستخدام أسلوب تحكم لكسب السيطرة على الناقل للفترة المتبقية من** ....

العملية (أي الاستيلاء على الناقل لطلب القراءة أو الكتابة ، ثم الاستيلاء على الناقل مرة أخرى لإجراء القراءة أو الكتابة) .



الشكل (3.21) - أنماط النقل لناقل البيانات

وفي الناقلات المخصصة (أحدها للعنوان والأخر للبيانات) يتم وضع العنوان **على متن ناقل العنوان ويستقر عليه** بينما يتم وضع البيانات على ناقل البيانات .

لعملية الكتابة يضع المُسيطر البيانات على ناقل البيانات **حالما يستقر العنوان** ويتمكن المستخدم من التعرف على العنوان . وفى **عملية القراءة يضع المستخدم** البيانات على ناقل البيانات بعدما يتعرف على العنوان ويجلب البيانات .

هناك تركيبة متنوعة من العمليات التى تسمح بها بعض الناقلات . **فعملية قراءة-** تعديل-كتابة عبارة عن عملية قراءة تتبعها فوراً كتابة **إلى نفس العنوان حيث** يعمم العنوان مرة واحدة فقط في بداية العملية . والعملية **برمتها هى عادةً غير** قابلة للتجزئة وذلك لمنع أي وصول إلى عنصر من البيانات **من قبل مسيطرين** أخرين محتملين على الناقل . والغرض الرئيسي من هذه **القدرة هو حماية الموارد** المشتركة من الذاكرة في الأنظمة ذات البرامج المتعددة .

قراءة-بعد-كتابة ، وهى عملية غير **قابلة للتجزئة تتكون من كتابة تليها مباشرة** قراءة من نفس العنوان ، وقد يتم **اجراء عملية القراءة لغرض التدقيق .**

بعض أنظمة الناقلات تدعم أيضا نقل البيانات بالقالب . **ففي هذه الحالة ، يتبع** دورة عنوان واحدة عددن من دورات البيانات **بحيث يتم نقل العنصر الأول من** البيانات من أو إلى عنوان محدد ، **ومن ثم تنقل عناصر البيانات المتبقية من أو** إلى عناوين لاحقة تسلسلياً .

مصطلحات مهمة

مسجل	Register
ثمان : 8 – خانات ثنائية	Byte
تعطيل المقاطعة	Disabled Interrupt
التحكيم الموزع	Distributed Arbitration
دورة التعليمة	Instruction Cycle
دورة التنفيذ	Instruction Execute
جلب التعليمة	Instruction Fetch
مقاطعة	Interrupt
مناول المقاطعة	Interrupt Handler
روتين خدمة المقاطعة	Interrupt Service Routine
مسجل عنوان الذاكرة	Memory Address Register (MAR)
مسجل الذاكرة المؤقت	Memory Buffer Register (MBR)
رابط المكونات الملحقة	Peripheral Component Interconnect (PCI)
عرض الناقل	Bus Width
تحكيم/مُحكم	Arbiter
وحدة الإدخال/الإخراج	I/O Module
مسجل عنوان الإدخال/الإخراج	I/O Address Register (I/O AR)
مسجل مؤقت الإدخال/الإخراج	I/O Buffer Register (I/O BR)
وحدة المعالجة الرئيسية	CPU
الذاكرة الرئيسية	Main Memory
دورة الجلب	Fetch Cycle
برنامج مادي	Hardwired Program
مجمع	Accumulator (AC)
حساب عنوان التعليمة	Instruction Address Calculation (IAC)
تفسير عمل التعليمة	Instruction Operation Decoding (IOD)
عملية على البيانات	Data Operation (DO)
تخزين المعامل	Operand Store (OS)
دورة المقاطعة	Interrupt Cycle
الوصول المباشر للذاكرة	Direct Memory Access (DMA)
اشعار/أقرار	Acknowledge
المشاركة الزمنية	Time Multiplexing
دورة الناقل	Bus Cycle
دورة نبضة الساعة	Clock Cycle
مسيطر (سيد)	Master

بيانات	Data
رموز التعليمة	Instruction Codes
اشارات التحكم	Control Signals
وظائف منطقية	Logic Functions
مفسر التعليمة	Instruction Interpreter
وظائف حسابية	Arithmetic Functions
البرمجة	Programming
برمجيات	Software
عتاد/كيان	Hardware
التوقيت المتزامن	Synchronous Timing
ناقل النظام	System Bus
التوقيت غير المتزامن	Asynchronous Timing
ناقل	Bus
تحكيم مركزى	Centralized Arbitration
ناقل البيانات	Data Bus
تخزين مؤقت	Buffer
عدد البرنامج	Program Counter (PC)
مسجل التعليمة	Instruction Register (IR)
وحدة التنفيذ	Execution Unit
دورة التنفيذ	Execute Cycle
الإدخال/الإخراج	Input/Output (I/O)
رمز العملية	Operation Code (Opcode)
جلب التعليمة	Instruction Fetch (IF)
حساب عنوان المعامل	Operand Address Calculation (OAC)
جلب المعامل	Operand Fetch (OF)
مؤقت	Timer
روتين خدمة المقاطعة	Interrupt Service Routine (ISR)
منفذ أدخال/إخراج	I/O Port
نظم ربط الحاسوب	SCSI (Small Computer System Interface)
التخصيص المادى	Physical Dedication
متحكم الناقل	Bus Controller
مستخدم (خادم)	Slave
قالب	Block
شعاع	Vector
شبكة محلية	Local Area Network (LAN)
شبكة موسعة	Wide Area Network (WAN)

أسئلة للمراجعة

1.- وضح مخطط الدورة التفصيلية للتعليمة مع بيان وصفى لكل حالة فى الدورة ؟

2.- وضح ماهى مكونات الحاسب و كيفية عملها معا ؟

3.- وضح بالرسم تسلسل التحكم فى برنامج بمقاطعة وبدون مقاطعة ؟

4.- ماهو تأثير المقاطعة على دورة التعليمة ؟

5.- ماهى المقاطعات المتعددة و ماهى اساليب حل هذه المشكلة ؟

6.- وضح ماهى انواع التبادلات التى تحتاج اليها الذاكرة للتواصل و العمل فى الحاسب ؟

7.- وضح ماهى انواع التبادلات التى يحتاج اليها المعالج للتواصل و العمل فى الحاسب ؟

8.- وضح ماهى انواع التبادلات التى تحتاج اليها وحدات الإدخال/الإخراج للتواصل و العمل فى الحاسب ؟

9.- لماذا معظم أنظمة الحاسب تستخدم ناقلات متعددة ؟

10.- ماهى عناصر تصميم الناقل ؟

11.- وضح ماهى معمارية ميزانين للناقلات ؟

12.- وضح ماهى بنية الناقل ولماذا معظم أنظمة الحاسب تستخدم ناقلات متعددة؟

13.- ماهى عناصر تصميم الناقل ؟

14.- المثال الموضح فى الشكل (3.5) يبين كيفية تنفيذ برنامج ما و حالة بعض المسجلات اثناء كل خطوة وتم وصفه فى 6 خطوات ، اذا اضفنا المسجلين (MAR) و (MBR) لوصف تنفيذ البرنامج ، ماهى محتوياتهما اثناء خطوات تنفيذ البرنامج (اعد رسم الشكل باضافة المسجلين مع المسجلات الموجودة بالشكل واستعمال هذين المسجلين فى خطوات تنفيذ البرنامج) ؟

15.- وضح بالرسم تسلسل الاحداث فى برنامج عند حدوث مقاطعة ؟

16.- افترض ان ثلاث مقاطعات (أ ، ب ، ج) قاطعت المعالج فى الازمنة التالية: (33 ث ، 35 ث ، 38 ث) حسب الترتيب ، و كانت اسبقياتها - الأعلى فالأدنى - (ب ، أ ، ج) و كل مقاطعة تستغرق 5 ثوانى من المعالج لتنفيذ الروتين الخاص بها ، **ماهو تسلسل تنفيذها من قبل المعالج فى حالة** تفعيل الاسبقيات او تعطيل الاسبقيات ؟

17.- افترض معالج له مسجل التعليمة بطول 16-خانة ومسجل عداد البرنامج بطول 10-خانة ، فما هو:

أ.- ماهو اقصى سعة للذاكرة القابلة للعنونة ؟

ب.- ماهو طول التعليمة (عدد الخانات)؟

ت.- ماهو عرض ناقل العناوين؟

ث.- ماهو عرض ناقل البيانات؟

ج.- ماهو عرض كل موقع بالذاكرة ؟

ح.- كم عدد الخانات المطلوبة لمسجل الذاكرة المؤقت (MBR)؟

18.- المعالج الافتراضى المذكور فى الشكل (3.4) **اضيفت له التعليمات التالية:**

0011 = Load AC from I/O

0110 = Store AC to I/O

فى هذه الحالة ، الجزء المكون من 12-خانة فى التعليمة والخاص بالعنوان يحدد جهاز أدخال/إخراج معين ، وضح كيفية تنفيذ البرنامج (على هيئة الجدول المبين بالمثال) للبرنامج التالى :

أ.- حُمل المجمع (AC) من الجهاز 5

ب.- اضف للمجمع محتويات الموقع 940

ت.- خزن محتويات المجمع فى الجهاز 6

19.- افترض أن معالج له 5 خطوط مقاطعة (0 – 4) ، **و بآلية تعطى** للمقاطعات الأدنى رقماً أسبقية فى المعالجة على **المقاطعات الأعلى رقماً،** المعالج بدأ العمل بدون أى مقاطعة فى الأنتظار ثم **حدث التسلسل التالى** من المقاطعات : 1 – 4 – 3 – 0 – 2 – 3 – 1 ، **مع أفتراض أن الزمن** المستغرق لمعالجة المقاطعة الواحدة **كافى لأن تصل فيه مقاطعتين أخريين** (أثناء معالجة مقاطعة تصل مقاطعتين) ، **وكل مقاطعة لا يمكنها أن تقاطع** الأخرى . حتى نهاية وصول **التسلسل المذكور من المقاطعات بالكامل ،** ماهو تسلسل تنفيذها من قبل المعالج؟

20.- أفترض أن معالج ما صدر منه عنواناً بعرض 16-خانة وله ناقل بيانات بعرض 16-خانة :

أ.- ماهي أقصي سعة للذاكرة يمكن للمعالج أن يتواصل معها أذا تم ربطه مع ذاكرة بعرض 16-خانة .

ب.-ماهي أقصي سعة للذاكرة يمكن للمعالج أن يتواصل معها أذا تم ربطه مع ذاكرة بعرض 8-خانة .

ت.-ماهى الخاصية المعمارية التي يمكن أن تسمح لهذا المعالج أن يتواصل مع مجموعة وحدات أدخال/إخراج منفصلة عن الذاكرة.

ث.-أذا كان للمعالج تعليمات أدخال/إخراج والحقل الخاص بتحديد رقم المنفذ في التعليمة عرضه 8-خانات ، كم عدد المنافذ ذات 8-خانات التي يمكن لهذا المعالج ان يتعامل معها . (وضح)

21.- المعالج أنتل 8088 له توقيتات ناقل قراءة مشابه لما هو موضح في المخطط المبين في الشكل (3.19) ولكنه يتطلب أربع دورات من نبضة المعالج ، البيانات الصحيحة موجودة على الناقل لفترة تصل حتى الدورة الرابعة لنبضة المعالج ، ومع أفتراض أن معدل نبضة ساعة المعالج هي 8 ميغاهيرتز :

أ.- ماهو المعدل الأقصى لنقل البيانات ؟

ب.- أذا تمت أضافة حالة انتظار واحدة لكل ثُمانٍ من البيانات يتم نقلها، فماهو المعدل الأقصى لنقل البيانات ؟

الفصل الرابع

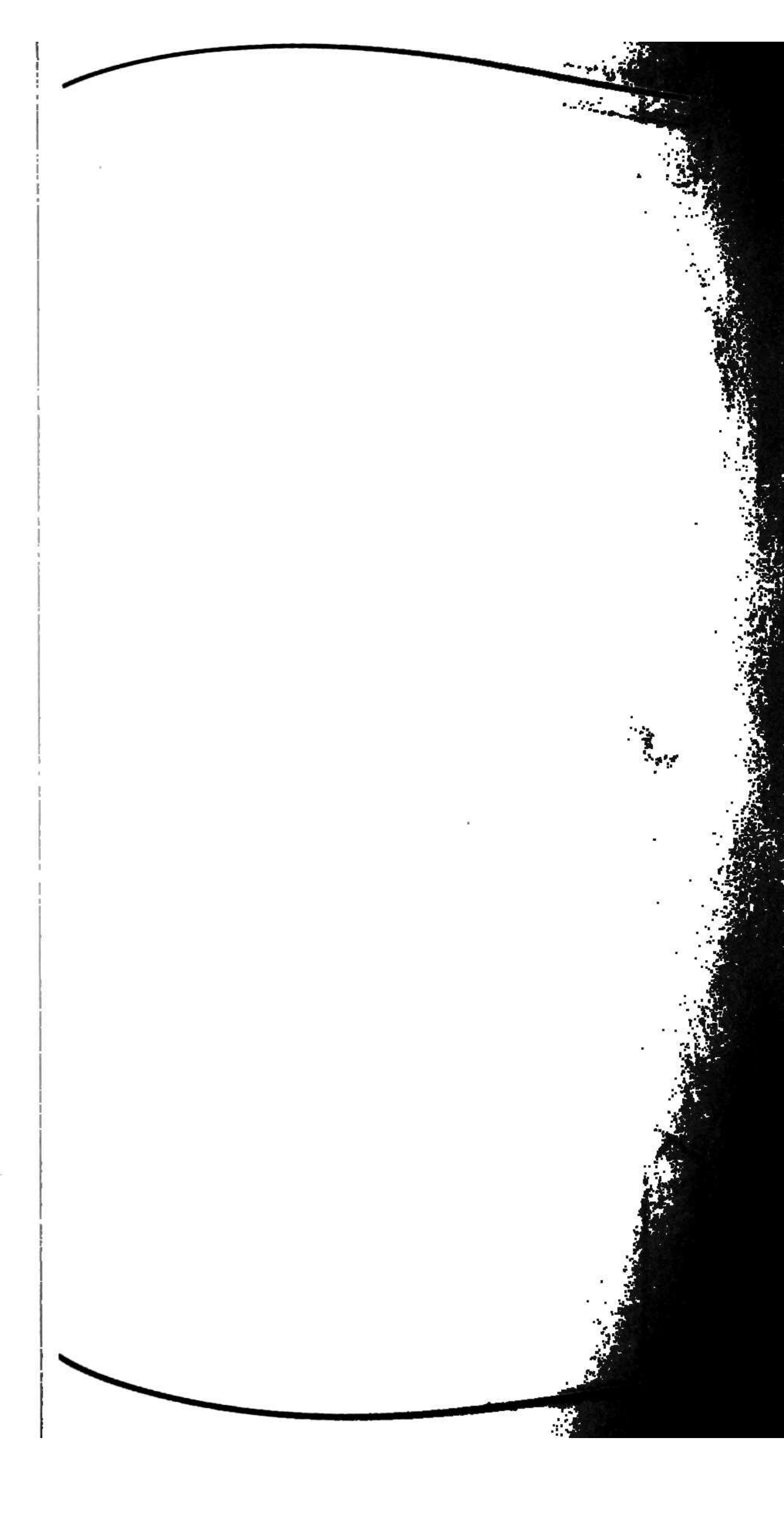
المعالجة الحسابية في الحاسب الآلي

4 - المعالجة الحسابية فى الحاسب

فى هذا الفصل سوف نستعرض بشكل عام وحدة الحساب والمنطق فى المعالج ، وسنركز على أهم وظيفة لوحدة الحساب والمنطق وهى الحساب الثنائى . الحساب الثنائى ينجز على نوعين مختلفين من الأعداد : الصحيح والحقيقى (النقطة العائمة) ، وفى الحالتين الطريقة المختارة لتمثيل الأعداد فى الحاسب هى قضية جوهرية فى التصميم . سنتعرض بالنقاش أيضاً للعمليات الحسابية الأساسية على الأعداد الثنائية الصحيحة بدون إشارة فقط .

4.1 وحدة الحساب والمنطق

وحدة الحساب والمنطق (ALU) هى ذلك الجزء من الحاسب الذي ينفذ العمليات الحسابية والمنطقية على البيانات ، وكل العناصر الأخرى المكونة لنظام الحاسب - وحدة التحكم والمسجلات والذاكرة و وحدة الإدخال/الإخراج - هى فى الأساس لإحضار البيانات إلى وحدة الحساب والمنطق لغرض المعالجة من ثم العودة بالنتائج . الشكل (4.1) يبين - بصورة عامة - كيفية ربط وحدة الحساب والمنطق مع باقى مكونات المعالج . يتم تقديم البيانات إلى وحدة الحساب والمنطق من المسجلات ، ويتم تخزين نتائج العملية في المسجلات . وهذه المسجلات هي أماكن تخزين مؤقتة داخل المعالج ترتبط بوحدة الحساب والمنطق بواسطة مسارات إشارة . وحدة الحساب والمنطق قد تضبط أعلام نتيجة لعملية ما ، فعلى سبيل المثال يتم رفع علم الفيض إلى (1) إذا كانت نتيجة العملية الحسابية تتجاوز طول المسجل الذى سيتم تخزين النتائج به . وتقدم وحدة التحكم الأشارات التي



تتحكم في عمل وحدة الحساب والمنطق ، وحركة البيانات من والى وحدة الحساب **والمنطق.**



الشكل (1.4) - مدخلات و مخرجات وحدة الحساب والمنطق

4.2 تمثيل الأعداد الصحيحة

ليس هناك داعى لإشارة السالب والفاصلة من أجل التخزين والمعالجة البسيطة **بالحاسب ، ويمكن** أستخدام الارقام الثنائية (0 و1) فقط لتمثيل الأعداد أذا كنا **محدودين بالأعداد الصحيحة** الغير سالبة ، التمثيل واضح ومباشر بحيث يمكن **لكلمة من 8-خانات ان تمثل الأرقام من 0 إلى 255** ، مثال ذلك :

11111111 = 255
10000000 = 128
00101001 = 41
00000001 = 1
00000000 = 0

بصفة عامة ، أى نسق مكون من n-خانة من الخانات الثنائية ويمثل علي شكل $a_{n-1}\, a_{n-2} \ldots a_1\, a_0$ **يمكن تفسيره** بالعدد الصحيح A وبدون إشارة و قيمته هي:

$$A = \sum_{i=0}^{n-1} 2^i a_i$$

4.2.1 تمثيل إشارة المقدار

أبسط شكل من أشكال التمثيل التي توظف خانة للإشارة **هو تمثيل إشارة المقدار.** في كلمة من ن– خانة ، الخانة الاقوى (أقصى اليسار) **في الكلمة هى خانة الإشارة** فأذا كانت خانة الإشارة 0 فالعدد موجب وإذا كانت خانة الإشارة 1 **فالعدد سالب،** بقية الخانات تحدد مقدار العدد الصحيح الذى يمثل العدد .

$-18_{10} = 10010010$ (إشارة المقدار)

$+18_{10} = 00010010$

هناك سلبيات عديدة لتمثيل إشارة المقدار منها أن **الجمع والطرح يتطلب مراعاة** إشارات الأعداد وكذلك مقادير ها لتنفيذ العملية المطلوبة ، **والعيب الآخر هو أن** هناك تمثيلين للصفر :

$-0_{10} = 10000000$ (إشارة المقدار)

$+0_{10} = 00000000$

وبصفة عامة يمكن وصفه بالصيغة التالية :

$$A = \begin{cases} \sum_{i=0}^{n-2} 2^i a^i & \text{if } a_{n-1} = 0 \\ -\sum_{i=0}^{n-2} 2^i a^i & \text{if } a_{n-1} = 1 \end{cases}$$

4.2.2 تمثيل المكمل الثانى

للتغلب على عيوب التمثيل السابق **وجد تمثيل المكمل الثانى ، حيث يوجد تمثيل واحد** للصفر وكذلك **يمكن أجراء العمليات الحسابية مباشرة بدون مراعاة لإشارة** العدد. **فتمثيل المكمل الثانى مثل تمثيل إشارة المقدار حيث يستخدم الخانة الاقوى**

(آخر خانة) كخانة إشارة مما يسهل اختبار ما إذا كان العدد موجب أو سالب ، ولكنه يختلف عنه فى كيفية تفسير بقية الخانات .

تمثيل الأعداد الصحيحة باستخدام تمثيل المكمل الثانى يتم بإضافة 1 الى المكمل الاول للعدد . لتوضيح ذلك دعونا ننظر للمثال التالى ، فكيف يمكننا أن نمثل العدد $(-5)_{10}$ بصورة المكمل الثانى مستخدمين 4-خانات ثنائية :

- **أولا ، علينا كتابة القيمة الموجبة للعدد ثنائي :** $(+5)_{10} = 0101$
- **التالى ، نعكس كل خانة للحصول على المكمل الاول :** 1010
- **أخيرا ، نضيف 1 إلى العدد السابق :** $1011 \rightarrow (-5)_{10}$

ويمكن اجمال خصائص المكمل الثانى فى الجدول التالى :

المدى	$-2^{\text{ن}-1} \;:\; 2^{\text{ن}-1}-1$
عدد مرات تمثيل الصفر	واحد
المعكوس	المعكوس المنطقى لكل خانة فى العدد الموجب، ثم معالجة العدد الناتج كعدد ثنائى صحيح بدون إشارة وذلك باضافة "1" له .
تمديد عدد الخانات	اضافة خانات الى اقصى اليسار و بقيمة خانة الإشارة الاصلية .
قاعدة الفيض	اذا تم جمع عددين بنفس الإشارة (سالبين او موجبين) ، يحدث الفيض اذا كان الناتج له إشارة مخالفة لهما .
قاعدة الطرح	لطرح *ص* من *س* ، ناخذ المكمل الثانى للعدد *ص* ثم نضيفه الى العدد *س* .

4.3 حساب الأعداد الصحيحة

4.3.1 النفي

القاعدة في تمثيل إشارة المقدار لنفي (معكوس) عدد صحيح بسيطة : عكس خانة **الإشارة . وفي التمثيل بالمكمل الثاني يمكن نفي عدد صحيح** بإتباع القاعدة التالية:

1. ايجاد المكمل المنطقي لكل خانة من العدد الصحيح بما في ذلك **خانة الإشارة،** بمعنى تغيير كل 1 الى 0 و كل 0 الى 1 .
2. معالجة النتيجة كعدد ثنائى صحيح بدون إشارة ، وإضافة 1.

ويشار إلى هذه العملية بعملية المكمل الثاني أو أخذ المكمل الثاني لعدد صحيح :

$(+18)_{10}$ = 00010010
(المكمل الاول) = 11101101
\+ 1
$(-18)_{10}$ = (المكمل الثانى) = 11101110

وكما هو متوقع معكوس النفي هو نفس العدد :

$(-18)_{10}$ = 11101110
(المكمل الاول) = 00010001
\+ 1
$(+18)_{10}$ = (المكمل الثانى) = 00010010

هناك حالتين يجب أخذهما فى الأعتبار، أولاً ، أذا كانت قيمة العدد *س*$= 0$ ، وفى حالة كلمة من 8-خانات :

$(0)_{10}$ = 00000000
(المكمل الاول) = 11111111
\+ 1
$(0)_{10}$ = (المكمل الثاني) = 1 00000000 ،

ويوجد حمل ناتج من الخانة الأقوى و تم تجاهله ، **والنتيجة هي أن نفي $(0)_{10}$ هو** $(0)_{10}$ وكما ينبغي أن يكون . الحالة الخاصة الثانية **وهى أكثر إشكالا ، فإذا أخذنا** معكوس لنمط من الخانات يبدأ بالرقم 1 **يتبعه اصفار بالكامل سوف نحصل على** نفس العدد ، **وعلى سبيل المثال كلمة من 8-خانات :**

$$(-128)_{10} = 10000000$$

$$\text{(المكمل الاول)} = 01111111$$

$$+ \quad 1$$

$$(-128)_{10} = \text{(المكمل الثاني)} = 10000000$$ ،

مثل هذا الشذوذ لا مفر منه .

4.3.2 الجمع و الطرح

الشكل (4.2) يوضح الجمع (الأضافة) باستخدام المكمل الثاني . الجمع يتم كما **لو أن العددان صحيحان** بدون إشارة ، والأربعة أمثلة توضح ذلك . فإذا كان ناتج **العملية موجب سوف** نحصل على عدد موجب في شكل صحيح ، أما اذا كان ناتج **العملية سالب سوف** نحصل على عدد سالب في شكل المكمل الثاني . نلاحظ أنه **في بعض الحالات يوجد** حمل ناتج (المشار إليه بواسطة التظليل) والذي يتم **تجاهله .**

في أي عملية جمع فإنه من الممكن ان يكون الناتج أكبر من حجم الكلمة المستخدمة **لحمله وهذا ما يسمى بالفيض .** عندما يحدث فيض ، يجب على وحدة الحساب **والمنطق ان تؤشر لهذا الأمر** (رفع علم الفيض) بحيث لا تتم أي محاولة لإستخدام **الناتج . للكشف عن الفيض ، يلتزم** بالقاعدة التالية :

قاعدة فيض : إذا تم جمع عددين ، وهما موجبين أو سالبين على حد سواء ، يحدث **الفيض فقط إذا كان الناتج ذو إشارة مخالفة لهما.**

أما الطرح فيتم بتباع القاعدة التالية (الشكل (4.3)) :

قاعدة الطرح : لطرح عدد واحد (المطروح - ط) من آخر (الطارح - م) نأخذ **المكمل الثانى للمطروح و نجمعه مع الطارح .**

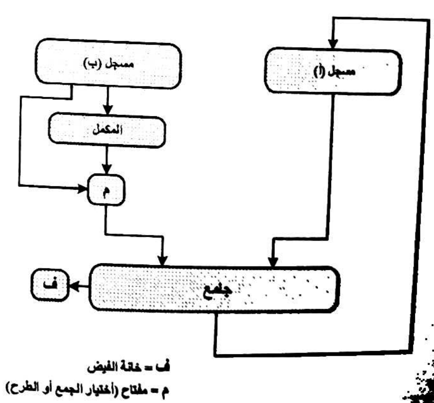
1001 = −7 +0101 = 5 1110 = −2 (ب) (−7) + (+5)	1100 = −4 +0100 = 4 10000 = 0 (أ) (−4) + (+4)
0011 = 3 +0100 = 4 0111 = 7 (د) (+3) + (+4)	1100 = −4 +1111 = −1 11011 = −5 (ج) (−4) + (−1)
0101 = 5 +0100 = 4 1001 = الفيض (ي) (+5) + (+4)	1001 = −7 +1010 = −6 10011 = الفيض (و) (−7) + (−6)

الشكل (4.2) – جمع الأعداد بصيغة المكمل الثانى

0010 = 2 +1001 = −7 1011 = −5 (أ) م = 2 = 0010 ط = 7 = 0111 -ط = 1001	0101 = 5 +1110 = −2 10011 = 3 (ب) م = 5 = 0101 ط = 2 = 0010 -ط = 1110
1011 = −5 +1110 = −2 11001 = −7 (ج) م = −5 = 1011 ط = 2 = 0010 -ط = 1110	0101 = 5 +0010 = 2 0111 = 7 (د) م = 5 = 0101 ط = −2 = 1110 -ط = 0010
0111 = 7 +0111 = 7 1110 = الفيض (و) م = 7 = 0111 ط = −7 = 1001 -ط = 0111	1010 = −6 +1100 = −4 10110 = الفيض (ي) م = −6 = 1010 ط = 4 = 0100 -ط = 1100

الشكل (4.3) - طرح الأعداد بصيغة المكمل الثانى

المخطط التصميمى الموضح فى الشكل (4.4) يبين مسار البيانات والعناصر **المادية اللازمة لتحقيق عملية الجمع والطرح** ، العنصر الرئيسي فى التصميم هو **الجامع الثنائي والذي يمكنه جمع عددين** وينتج عنه حاصل الجمع و إشارة الفيض. **ويعامل الجامع الثنائي العددين** على أنهما عددين صحيحين بدون إشارة، ولإنجاز **عملية الجمع يتم تقديم العددين** إلى الجامع من مسجلين وهما في هذه الحالة **المسجلين (أ) و (ب)** ، والنتيجة قد يتم تخزينها في إحدى هذه المسجلات أو في **مسجل ثالث ، وخانة الفيض** فى التصميم يخزن بها مؤشر/علم الفيض (0 = لا **يوجد فيض ، 1 = يوجد فيض)** . أما فى الطرح ، يتم تمرير المطروح (المسجل **(ب)) الى المكمل الثانى** بحيث يتم تقديمه للجامع فى صيغة مكمل الثانى ، لاحظ **أن الشكل لا يظُهر** إلا مسارات البيانات ، ولذلك هناك حاجة إلى إشارات تحكم **للتحكم فى إدخال البيانات** بحيث تمر عبر المكمل أم لا وهذا يتوقف على ما إذا **كانت العملية جمع أو طرح.**



-مكونات الكيان المادى الخاص بعمليتي الجمع والطرح

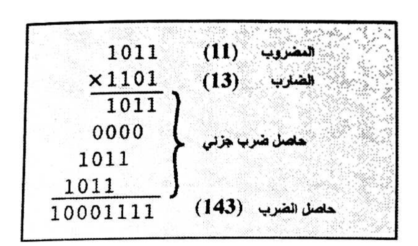
4.3.3 الضرب

بالمقارنة مع عملية الجمع والطرح ، الضرب **عملية معقدة سواء أجريت بالكيان** المادى أو برنامج ، وقد أستخدمت مجموعة واسعة من **الخوارزميات لهذا** الغرض . سنبدأ مع ضرب عددين صحيحين بدون إشارة ومن ثم ننظر في إحدى التقنيات الأكثر شيوعا لضرب الأعداد الصحيحة بدون إشارة (موجبة) .

ضرب الأعداد الصحيحة بدون إشارة

الشكل (4.5) يوضح ضرب الأعداد الثنائية الصحيحة بدون إشارة ، وعند اجرائها باستخدام الورقة والقلم ، يمكن ملاحظة التالى :

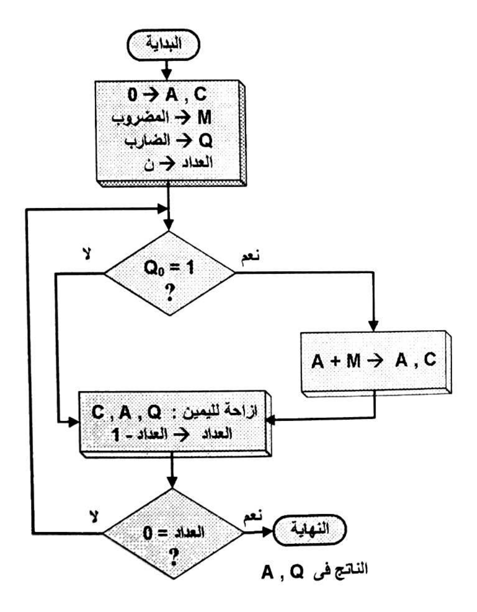
1. ينطوي الضرب على أنتاج عدد من النتائج الجزنية ، **واحدة لكل خانة في** العدد الضارب وهذه النتائج الجزنية تُجمع للحصول على الناتج النهائي .
2. النتائج الجزنية يكمن إيجادها بسهولة ، فعندما تكون خانة الضارب 0 ، الناتج الجزني 0 ، عندما تكون خانة الضارب 1 ، **فالناتج الجزئي هو** المضروب والذى سيكون 1 أو 0 ، وذلك لإنه فى الحساب **الثنائى حاصل** ضرب أى عدد فى 1 هو العدد نفسه (نسخة منه) .
3. ويتم إيجاد الناتج الأجمالي عن طريق جمع النتائج الجزنية . **ولهذه العملية** يتم إزاحة الناتج الجزنى التالى خانة واحدة إلى **اليسار نسبة الى الناتج** الجزئي السابق له ، ولاحظ هنا إنه تُجري **عملية إزاحة للنتائج الجزنية بناءً** على قوة الخانة الضاربة فى العدد المضروب .
4. ضرب عددين ثنائيين صحيحين بعدد *ن- خانة* **ينتج عنه ناتج بطول يصل الى** *2ن- خانة* . (مثال : $(1001)_2 = 11 \times 11)$) .



الشكل (4.5) – ضرب الأعداد الثنائية الصحيحة بدون إشارة

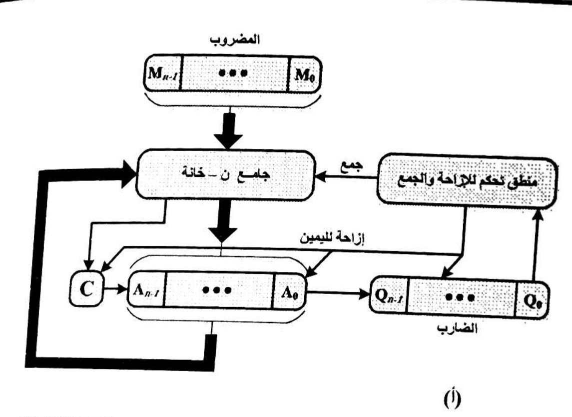
بالمقارنة مع نهج الورقة والقلم هناك عدة أشياء يمكننا القيام بها لجعل الضرب **المحوسب أكثر** كفاءة . أولا ، يمكننا إجراء عملية جمع سريعة للنتائج الجزئية **بدلا من الأنتظار** حتى النهاية وهذا يلغي الحاجة لتخزين كل النتائج الجزئية مما **يقلل** عدد المسجلات المطلوبة . وثانيا ، يمكننا أختصار بعض الوقت فى إيجاد **النتائج الجزئية** ، فلكل 1 من العدد الضارب نحتاج الى عملية جمع و أزاحة ، **ولكن لكل 0 نحتاج** فقط عملية أزاحة ، والمخطط الأنسيابي المبين في الشكل **(4.6) يوضح** خوارزمية عملية الضرب . فالضارب و المضروب يتم تحميلهما **فى المسجلين (Q) و** (M) ، نحتاج أيضا لمسجل ثالث وهو المسجل (A) ويتم **في البداية تهيئته** إلى 0 . وهناك أيضا المسجل (C) وهو بطول خانة واحدة و **يهيئ فى** البداية الى 0 وذلك لتخزين الحمل الناتج والمحتمل من عملية الجمع . **عملية الضرب هى** كما يلي : منطق التحكم يقرأ أجزاء الضارب خانة بخانة ، **إذا كانت الخانة $(Q)_0$** هى 1 ، وتتم اضافة المضروب إلى المسجل (A) ويتم **تخزين النتيجة في المسجل** (A) مع استخدام خانة الفيض (C) ، من ثم يتم إزاحة **كافة خانات المسجلات** (C) و (A) و (Q) خانة واحدة لليمين بحيث تزاح الخانة **إلى $(A)_{ن-1}$ ،** $(A)_0$ **تذهب إلى** $(Q)_{ن-1}$ **وتُفقد الخانة** $(Q)_0$.

إذا كانت الخانة $(Q)_0$ تساوى 0 فلا تتم عملية إضافة بل مجرد **إزاحة لكافة** المسجلات (كالإزاحة السابقة) . وتتكرر هذه العملية **لكل خانة من الضارب** الأصلي . الناتج بعدد *2ن* من الخانات يخزن فى المسجلات (A) و (Q) .



الشكل (4.6) – المخطط الأنسيابى لعملية **ضرب الأعداد الثنائية الصحيحة** بدون إشارة

والشكل (4.7) يبين الكيان المادي **لعملية الضرب مع مثال يوضح العملية ،** ولاحظ أنه في الدورة الثانية ، **عندما تكون خانة المضروب 0 ليس هناك عملية** إضافة .



(أ)

	C	A	Q	M
قيم ابتدائية	0	0000	1101	1011
الدورة الأولى – جمع	0	1011	1101	1011
الدورة الأولى – إزاحة	0	0101	1110	1011
الدورة الثانية – إزاحة	0	0010	1111	1011
الدورة الثالثة – جمع	0	1101	1111	1011
الدورة الثالثة – إزاحة	0	0110	1111	1011
الدورة الرابعة – جمع	1	0001	1111	1011
الدورة الرابعة – إزاحة	0	1000	1111	1011

(ب)

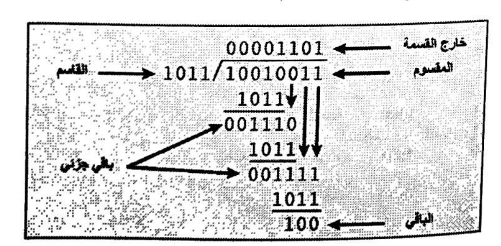
الشكل (4.7) – الكيان المادى ومثال لعملية ضرب الأعداد الثنائية الصحيحة بدون إشارة

المخطط الأنسيابي السابق يبين بوضوح أن الضرب لن يعمل إذا كان المضروب **سالباً . هناك عدد من الطرق** للخروج من هذه المعضلة ، وأحدها وأكثرها شيوعا **هي خوارزمية "بووث"** .

وهذه الخوارزمية لديها ميزة تسريع عملية الضرب **مقارنة بالطريقة التقليدية**.

4.3.4 القسمة

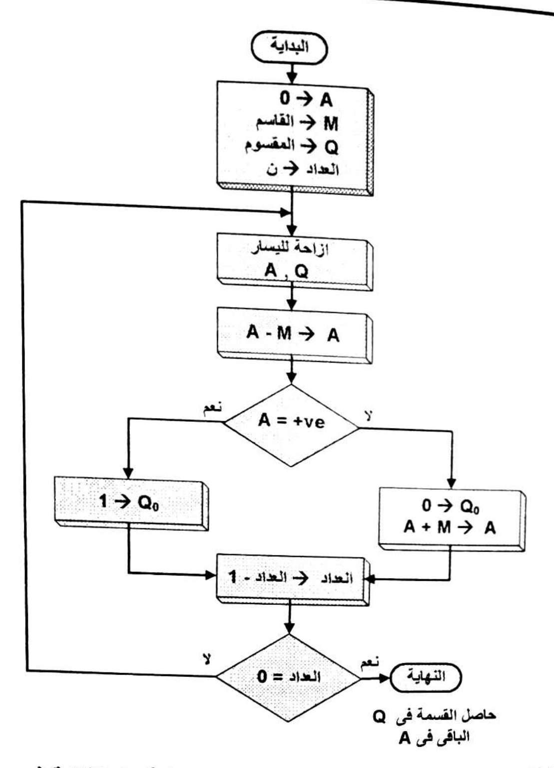
التقسيم هو نوعا ما أكثر تعقيداً من الضرب ولكنه مبني على نفس المبادئ العامة. وكما سبق ، فأن أساس الخوارزمية هو نهج الورقة والقلم ، وهذه العملية تنطوي على الأزاحة المتكررة والجمع أو الطرح . الشكل (4.8) يوضح مثالا للتقسيم المطول للأعداد الصحيحة بدون إشارة ، ومن المفيد هنا أن نصف العملية بالتفصيل . أولا ، يتم فحص أجزاء المقسوم من اليسار إلى اليمين حتى تمثل مجموعة الخانات التي تم فحصها عدداً أكبر من أو يساوي القاسم ، و يشار لهذا على أن القاسم قادر على تقسيم العدد ، وقبل هذه اللحظة يوضع 0 فى الناتج من اليسار لليمين ، وعند إجراء هذا الحدث يتم وضع 1 في الحاصل ويتم طرح القاسم من المقسوم وينتج عنه ناتج جزئي ، ويشار إلى الناتج على أنه الباقى الجزئي . ومن هذه النقطة التقسيم يتبع نمطا دورياً ، ففي كل دورة يتم إلحاق خانة إضافية من المقسوم على الباقى الجزئي حتى يكون أكبر من أو يساوي القاسم . وكما سبق ، يتم طرح القاسم من هذا العدد لإنتاج باقي جزئي جديد . وتستمر العملية حتى يتم إستنفاد جميع خانات المقسوم .



الشكل (4.8) – مثال على التقسيم الأعداد الثنائية الصحيحة بدون إشارة

الشكل (4.9) يظهر آلية الخوارزمية التي توافق عملية القسمة المطولة ، وفيها يتم وضع القاسم في المسجل (M) ، والمقسوم في المسجل (Q) ، في كل خطوة يتم ازاحة المسجلين (A) و (Q) معا لليسار خانة واحدة ، يتم طرح (M) من (A) لتحديد ما إذا كان يقبل القسمة على الناتج الجزئى ، إذا حدث ذلك $(Q)_0$ تساوى 1 ، وغير ذلك فإن $(Q)_0$ تساوى 0 ويجب اعادة اضافة (M) الى (A) لإسترجاع قيمتها السابقة ، وينقص العداد وتستمر العملية لعددس من الخطوات، ولاحظ ان عدد الخطوات هو مساوى لعدد خانات كل عدد ولذلك يجب تعديل عدد خانات العددين لكى تكون متساوية قبل إجراء العملية كذلك فإن العددين ممثلين بصيغة المكمل الثانى ، وفى النهاية الحاصل يوجد فى المسجل (Q) والباقى فى المسجل (A) . والجدول التالى يوضح مثال (تقسيم $(7/3)_{10}$) على كيفية عمل خوارزمية التقسيم بالمكمل الثانى .

A	Q	مثال على القسمة : (تقسيم $(7/3)_{10}$) ، M=0011
0000	0111	القيم المبدئية
0000	1110	إزاحة
1101		للطرح نستخدام المكمل الثانى للعدد 0011
1101		
0000		طرح
	1110	ضع $Q_0 = 0$ ، أعادة قيمة A
0001	1100	إزاحة
1101		
1110		
0001		طرح
	1100	ضع $Q_0 = 0$ ، أعادة قيمة A
0011	1000	إزاحة
1101		
0000		
	1001	ضع $Q_0 = 1$ ، طرح
0001	0010	إزاحة
1101		
1110		
0001		طرح
	0010	ضع $Q_0 = 0$ ، أعادة قيمة A



الشكل (4.9) – المخطط الأنسيابى لعملية القسمة للأعداد الثنائية الصحيحة بدون إشارة

4.4 تمثيل النقطة العائمة

التمثيل بالنقطة الثابتة (الأعداد الصحيحة) له حدود ، فلا يمكن أن يُمثل الأعداد الكبيرة جداً ، كذلك لا يمكنه ان يمثل كسور صغيرة جداً في خانات محدودة . ومثال على ذلك يمكن للعدد 976،000،000،000،000 أن يمثل على شكل

9.67×10^{14} وكما يمكن للعدد 0.0000000000000976 أن يُمثل على شكل **9.67×10^{-14} وهذا يسمح بتمثيل مجموعة من الأعداد الكبيرة جداً والصغيرة جداً في أعداد قليلة فقط** (خانات محدودة) . والهدف من تمثيل النقطة العائمة :

1. **تصغير حيز** التمثيل الأعداد .
2. **تمثيل** الإعداد الكبيرة جداً بشكل عملي .
3. **تمثيل** الإعداد الصغيرة جداً بشكل عملي .
4. **الدقة في** العمليات الحسابية وتقليل الأخطاء التراكمية.
5. **التوفير في** الذاكرة .

أي عدد بصيغة النقطة العائمة يكتب على شكل خانات الكسر **(S=Significand)** ، وخانات الأس (E=Exponent) ، والقاعدة **(B=Base)** ، ويكتب العدد وفق العلاقة التالية (القاعدة = 2 للعدد الثنائي):

$$\pm S \times B^{\pm E}$$

والشكل (4.10) يوضح المبادئ المستخدمة في تمثيل العدد الثنائي بالنقطة العائمة **مع مثال على ذلك** ، ويظهر في الشكل نموذج لتنسيق عدد بتمثيل النقطة العائمة **بطول 32 – خانة** ، وأقصى خانة لليسار في التمثيل خاصة لإشارة الكسر **(0 = موجب، 1 = سالب)** ، وقيمة الأس تخزن في 8 – خانات التالية وباقي 23 **خانة للكسر.**

الأعداد التالية متكافئة ، حيث تم التعبير عن الكسر على شكل ثنائي :

0.0110×2^{6}
110×2^{2}
0.110×2^{5}

إشارة المقدار

8 خانة — 23 خانة

أس معدل | المقدار

(أ) - الشكل

$1.1010001 \times 2^{10100}$ = 0 10010011 10100010000000000000000 = 1.6328125×2^{20}
$-1.1010001 \times 2^{10100}$ = 1 10010011 10100010000000000000000 = -1.6328125×2^{20}
$1.1010001 \times 2^{-10100}$ = 0 01101011 10100010000000000000000 = 1.6328125×2^{-20}
$-1.1010001 \times 2^{-10100}$ = 1 01101011 10100010000000000000000 = $-1.6328125 \times 2^{-20}$

(ب) - الامثلة

الشكل (4.10) – نموذج لتنسيق عدد بالنقطة العائمة بطول 32 – خانة

تمثيل النقطة العائمة للإعداد بصيغة IEEE-754

من أشهر تمثيلات النقطة العائمة للأعداد هى صيغة IEEE-754 أحادية الدقة [32 – خانة] ، فى هذه الصيغة يتم حفظ العدد فى 32 – خانة ثنائية فى نظام الحاسب ، ويتم تمثيل العدد على شكل الصيغة القياسية التالية :

(-1)^س 1.ص × 2^ع

حيث س= إشارة العدد ، فإذا موجب = 0 وإذا سالب = 1 ، ص قيمة الكسر ، و ع قيمة الأس مضاف أليها 127 .

صيغة IEEE-754 احادية الدقة [32 – خانة] تُحدد خانة واحدة لأشارة العدد (س) ، و تحدد 8 – خانات للأس ، و اخيراً ، 23 – خانة للكسر (ص) ، **بحيث يكون تمثيل العدد فى 32 – خانة ثنائية بالترتيب التالى لكل مكونات الصيغة القياسية** : [[ص(23)][ع(8)][س(1)]] .

لتمثيل إى عدد ثنائى بهذه الصيغة يجب أولا تعديله حتى يصبح فى الصيغة القياسية (تعريف ص) وذلك بأن يبدأ العدد بخانة واحدة صحيحة بعد الفاصلة

(النقطة) قيمتها 1 و تزاح أى خانات صحيحة آخرى (أن وجدت) لبعد الفاصلة ، **وتتم إزاحة خانات العدد يميناً أو يساراً حتى نتحصل على الصيغة القياسية** (تحديد ص) ، **ويصاحب أزاحة خانات العدد تغيير فى قيمة الأس لأساس نظام العدد الثنائى - 2 (تتغيير قيمة ع)** ، فإذا كانت الأزاحة لليمين تزداد قيمة الأس و إذا **كانت لليسار تنقص قيمة الأس** ، وتحدد قيمة ع بأضافة 127 لقيمة الأس الناتج **بعد تعديل العدد و يصبح فى الصيغة القياسية** .

لنأخذ مثال على تمثيل الإعدد بالصيغة المذكورة وذلك بتمثيل العدد $(-5.25)_{10}$،
أولا ، العدد سالب ، إذن قيمة س = 1 ، والخانة الاولى فى التمثيل هى [1] .
وثانيا ، قيمة العدد بالنظام الثنائى هو $(101.01)_2$ ، لاحظ هنا أن العدد يبدأ بخانة **قيمتها 1 صحيح** وتوجد ثلاث خانات بعد النقطة ، لذلك يجب تعديل العدد ليصبح **فى الصيغة القياسية (ص.1)** و ما يصاحب ذلك من تعديل قى قيمة الأس، وبعد **أزاحة العدد خانتين لليسار** يصبح $(1.0101\times2^{2})_2$ ، وهذا العدد مساوى فى **القيمة للعدد** $(101.01)_2$ ولكن مع تغيير فى الشكل ، وبهذا تحصلنا على قيمة **الكسر** (ص = $(0101)_2$)) ، وتحصلنا على القيم التى ستوضع فى 23 – خانة **المخصصة للكسر** بحيث أن قيمة ص تكون الخانات الأولى فى 23 – خانة **المخصصة لتمثيل الكسر** [01010000000000000000000] .

اخيراً ، لإيجاد قيمة ع نضيف 127 لقيمة الأس الناتج بعد تعديل العدد للصيغة **القياسية** ، ع = 2 + 127 = 129 ، تمثيل قيمة الأس (ع) كعدد ثنائى يصبح $(10000001)_2$ ، وبهذا تحصلنا فى القيم التى ستوضع فى 8 – خانات **المخصصة للأس** [10000001] .

بتعريف قيمة س ، و ص ، و ع ، الأن يمكن كتابة العدد بتمثيل النقطة العائمة بصيغة IEEE-754 أحادية الدقة [32 – خانة] ، على الشكل التالى :

[11000000101010000000000000000000] .

يكمننا ان نلخص الإجراءات السالفة في المثالين التاليين :

مثال (1) : العدد $(-353.625)_{10}$: –

أولا ، تحويل العدد الى النظام الثنائى – $(-101100001.101)_2$

ثانيا ، العدد سالب ، س = $(1)_2$

ثالثا ، تعديل العدد للصيغة القياسية ، $(1.01100001101\times2^{8})_2$

رابعا ، بعد التعديل ، ص = $(01100001101)_2$

خامسا ، إيجاد قيمة الأس بالثنائى ، 8 + 127 = 135 ، ع = $(10000111)_2$

سادسا ، تمثيل العدد بتجميع عناصره السابقة :

[11000011101100001101000000000000] .

مثال (2) : العدد $(0.09375)_{10}$: –

أولا ، تحويل العدد الى النظام الثنائى – $(0.00011)_2$

ثانيا ، العدد موجب ، س = $(0)_2$

ثالثا ، تعديل العدد للصيغة القياسية ، $(1.1\times2^{-4})_2$

رابعا ، بعد التعديل ، ص = $(1)_2$

خامسا ، إيجاد قيمة الأس بالثنائى ، (4-) + 127 = 123 ،

ع = $(01111011)_2$

سادسا ، تمثيل العدد بتجميع عناصره السابقة :

[00111101110000000000000000000000] .

مصطلحات مهمة

القاسم	Divisor
الأس	Exponent
الفيض	Overflow
الجزئى	Partial
النقطة العائمة	Floating-point
المطروح	Minuend
الضارب	Multiplier
المعكوس	Negative
المكمل	Complement
إشارة المقدار	Sign-Magnitude
الحاصل	Quotient
المضروب	Multiplicand
عد	Count
حساب	Arithmetic
ازاحة	Shift
قاعدة	Base
حمل	Carry
تمثيل	Representation
المكمل الاول	One's Complement
المقسوم	Dividend
الكسر	Significand
المكمل الثانى	Two's Complement
الباقى	Remainder
الطارح	Subtrahend
وحدة الحساب والمنطق	Arithmetic & Logic Unit (ALU)
أعلام	Flags

أسئلة للمراجعة

-1. أشرح كيفية تحديد ما إذا كان العدد سالبا أم لا في التمثيلات التالية: المكمل الثانى ، إشارة المقدار.

-2. ما هي العناصر الأساسية للعدد في تمثيل النقطة العائمة ؟

-3. افترض ان الاعداد التالية بتمثيل عدد صحيح بدون أشارة ثم بتمثيل المكمل الثانى ثم بتمثيل أشارة المقدار ، أوجد صيغته العشرية لكل تمثيل : $(11101011)_2$ ،$(00101101)_2$ ،$(11101010)_2$.

-4. افترض ان الاعداد ممثلة في 8-خانات ، أحسب ما يلي (وضح الخطوات) $(55+77)_{10}$ ، $(-100+28)_{10}$ ، $(127-25)_{10}$.

-5. اوجد ناتج عملية الطرح التالية بإستخدام حساب المكمل الثانى:

1110111	110011
1001100-	011110-

-6. أضرب $(1010)_2$ فى $(111)_2$ بإستخدام المخطط الأنسيابي الموضح في الشكل (4.6) ، وضح الخطوات كما فى المثال المبين.

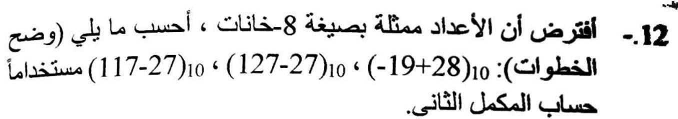
-7. أوجد حاصل قسمة $(45)_{10}$ على $(15)_{10}$ بالتمثيل الثانى مستخدماً المخطط الأنسيابى للقسمة المبين في الشكل (4.9) ، وضح الخطوات كما فى المثال المبين لذلك .

-8. وضح مكونات الكيان المادى الخاص بعملية الجمع والطرح مع شرح آلية عمله ؟

-9. مثل الأعداد العشرية التالية ثنائيا باستخدام 16 خانة مكمل ثانى و إشارة المقدار(وضح الخطوات) : $(377)_{10}$ ، $(445)_{10}$ ، $(-109)_{10}$.

-10. **أوجد حاصل قسمة $(75)_{10}$ على $(20)_{10}$ بالتمثيل الثانى بصورة المكمل الثانى مستخدما المخطط الانسيابى للقسمة الموضح في الشكل (4.9) والمثال الموضح لذلك.**

-11. **وضح مكونات الكيان المادى الخاص بعملية الضرب للعدد الثنائى؟**



.12- **أفترض أن الأعداد ممثلة بصيغة 8-خانات ، أحسب ما يلي (وضح الخطوات):** $(-19+28)_{10}$ ، $(127-27)_{10}$ ، $(117-27)_{10}$ مستخداماً حساب المكمل الثانى.

.13- وضح كيفية شكل العدد $(-5.325)_{10}$ ثنائياً ممثلاً بالنقطة العائمة بصيغة IEEE-754 ؟

.14- أضرب $(7)_{10}$ فى $(13)_{10}$ بإستخدام المخطط الأنسيابي الموضح في الشكل (4.6) ، وضح الخطوات كما فى المثال المبين لذلك .

الفصل الخامس

معمارية طقم التعليمات

5 - معمارية طقم التعليمات

طقم أو فئة تعليمات المعالج هى أحد الأوجه التى قد ينظر اليها كلّ من مصمم الحاسب و مبرمج الحاسب برؤية مختلفة . فمن وجه نظر المصمم طقم تعليمات المعالج يحدد الوظائف المطلوبة من المعالج و كيفية أنجازها وتصميمها (رؤية تنظيمية) ، فى حين أن المبرمج يهتم ببنية المسجلات والذاكرة وكذلك أنواع البيانات التى يمكن للمعالج أن يدعمها و وظائف وحدة الحساب والمنطق (رؤية معمارية) . إن وصف طقم تعليمات المعالج يمتد إلى شرح المعالج من الناحية التنظيمية ولكننا فى هذا الفصل سوف نقتصر على شرح خصائص طقم تعليمات المعالج و وظائفها كدراسة لوحدة التحكم فى المعالج من الناحية المعمارية .

5.1 خصائص تعليمات المعالج

يتحدد عمل المعالج من خلال التعليمات التي يمكنه تنفيذها ، ويشار إليها بتعليمات المعالج أو تعليمات الحاسب ويشار إلى مجموعة التعليمات المختلفة التي يمكن للمعالج أن ينفذها على أنها طقم أو فئة تعليمات المعالج .

5.1.1 عناصر تعليمة المعالج

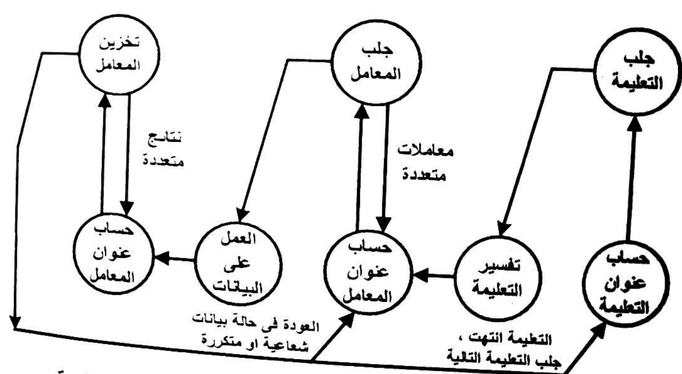
يجب على كل تعليمة أن تحتوي على المعلومات المطلوبة من قبل المعالج للتنفيذ. الشكل (5.1) يبين الخطوات المتبعة لتنفيذ التعليمة و – ضمنيا – يحدد عناصر التعليمة ، وهذه العناصر هي كما يلي:

1. **رمز العملية** : **يُحدد العملية التي يتعين القيام بها** (على سبيل المثال ، **جمع أو أدخال/إخراج) . فالعملية** يرمز لها من خلال رمز ثناني ، **ويطلق عليه رمز العملية .**

2. **مؤشر للمعامل المصدر** : أى عملية قد تنطوي على معامل مصدر أو اكثر، **وهذه المعاملات** هي المدخلات لهذه العملية .

3. **مؤشر للمعامل الناتج** : العملية قد تُسفر عن ناتج يجب أن يحفظ .

4. **مؤشر للتعليمة التالية** : يخبر المعالج من أين تُجلب التعليمة التالية بعد إنهاء **تنفيذ التعليمة** الحالية .



الشكل (5.1) – شكل تخطيطى للدورة المبسطة لحالات التعليمة

عنوان التعليمة التالية التي سيتم جلبها يمكن أن يكون إما عنواناً حقيقياً أو عنواناً **ظاهرياً حسب المعمارية** ، وهذا التمييز عادة ما يكون واضحاً فى معمارية طقم **التعليمات . وفي معظم الحالات التعليمة التالية التى سيتم جلبها هى تلقائيا بعد**

التعليمة الحالية ، وفى هذه الحالات لا يوجد مؤشر صريح إلى عنوان التعليمة التالية . **وعندما يكون هناك حاجة لمؤشر صريح لعنوان التعليمة التالية يجب أن** يتم تقديم موقعها بالذاكرة الرئيسية أو الظاهرية.

المعامل المصدر أو الناتج يمكن أن يكون فى أحد المناطق التالية :

1. الذاكرة الرئيسية أو الظاهرية : **كما هو الحال مع مؤشر التعليمة التالية ،** ويجب أن يتم توفير عنوان موقعه فى الذاكرة الرئيسية أو الظاهرية .
2. مسجل المعالج : مع أستثناءات نادرة **يحتوي المعالج على مسجل أو أكثر** يمكن الإشارة اليها بواسطة تعليمات المعالج . **فإذا كان هناك مسجل واحد فقط** الإشارة اليه تكون ضمنية **أما اذا كان هناك أكثر من مسجل فيجب تعريف أسم** أو رقم وحيد لكل مسجل والتعليمة **يجب أن تحتوي على رقم أو أسم المسجل** المطلوب (الذى يوجد به المعامل) .
3. فوري : قيمة المعامل **موجودة في حقل بالتعليمة الجارى تنفيذها .**
4. وحدة الإدخال/الإخراج : التعليمة **يجب أن تحدد وحدة الإدخال/الإخراج التى** يوجد بها معامل العملية .

5.1.2 تمثيل التعليمات

فى نظام الحاسب يتم تمثيل التعليمة بسلسلة من الثنائيات (**خانات ثنائية) . وتقسم** التعليمة الى عدة حقول ، ولكل عنصر مُكون للتعليمة **حقل يختص به والشكل** (5.2) يبين مثال بسيط لشكل أو تنسيق تعليمة . **فى أغلب أطقم التعليمات يتم** أستخدام أكثر من شكل واحد للتعليمة . **أثناء تنفيذ التعليمة يتم قراءة التعليمة** لمسجل التعليمة (IR) في المعالج ، **ومن ثم يجب على المعالج أن يكون قادراً** على أستخراج البيانات من **حقول التعليمة المختلفة لتنفيذ العملية المطلوبة.**

6 خانات ثنائية	6 خانات ثنائية	4 خانات ثنائية
مؤشر المعامل	مؤشر المعامل	رمز التعليمة

← 16 خانة ثنائية →

الشكل (5.2) - تنسيق لتعليمة بسيطة

من الصعب على المبرمج والقارئ على حد سواء التعامل مع التمثيل الثنائي **لتعليمات المعالج** ، وبالتالي فقد أصبح من الشائع استخدام التمثيل الرمزي **لتعليمات المعالج** . ومثال على ذلك طقم التعليمات المستخدمة مع نموذج الحاسب **(IAS) .** فرمز العملية عبارة عن أختصار رمزى يشير للعملية المزمع اجراءها، **ومن الأمثلة الشائعة** لرموز العمليات ما يلي:

ADD جمع
SUB طرح
MUL ضرب
DIV قسمة
LOAD حمّل بيانات من الذاكرة
STOR خزّن بيانات فى الذاكرة

المعاملات تمثل رمزياً ، فعلى سبيل المثال التعليمة [ADD R , Y] قد تعني **إضافة القيمة الواردة** في موقع البيانة Y إلى محتويات المسجل R ، وفي هذا **المثال Y تشير** إلى عنوان موقع في الذاكرة ، و R تشير إلى مسجل معين ، **ولاحظ أن العملية** تُنفذ على محتويات الموقع وليس على عنوانه .

مما سبق يتضح أنه من الممكن كتابة برنامج بلغة الآلة (المعالج) بشكل رمزي ، **وكل رمز عملية** لديه تمثيل ثنائي ثابت ، والمبرمج يحدد موقع المعامل الذى **يعمل معه التعليمة .**

5.1.3 أنواع التعليمات

ينبغي أن يكون للحاسب طقم من التعليمات التي تسمح للمستخدم/**المبرمج بصياغة** أي عمل/برنامج لمعالجة البيانات ، وبطريقة أخرى أي برنامج **مكتوب بلغة راقية** المستوى يجب أن يترجم إلى لغة الآلة ليتم تنفيذه ، وبالتالي **يجب على مجموعة** تعليمات المعالج ان تكون كافية للتعبير عن أي تعليمة **من اللغة الراقية المستوى.** ويمكننا تصنيف أنواع التعليمات على النحو التالي :

- معالجة البيانات : تعليمات الحساب والمنطق.
- تخزين البيانات : حركة البيانات ألى داخل أو **خارج المسجل و/أو** مواقع بالذاكرة .
- حركة بيانات : تعليمات الإدخال/الإخراج.
- التحكم : تعليمات الأختبار والتفرع.

التعليمات الحسابية توفر القدرات الحسابية **لمعالجة البيانات الثنائية . والتعليمات** المنطقية (بولي) تعمل على أجزاء من الكلمة الواحدة **كخانات ثنائية بدلا من** أعداد، وبالتالي فإنها توفر قدرة لمعالجة أي نوع آخر من **البيانات التي قد يرغب** المستخدم في اجراءها . وتُجرى هذه العمليات على **البيانات في مسجلات المعالج،** ولذلك يجب أن تكون هناك تعليمات تتعامل مع **الذاكرة لنقل البيانات بين الذاكرة** والمسجلات . وهناك حاجة أيضا إلى **تعليمات أدخال/إخراج لنقل البرامج** والبيانات إلى الذاكرة والعودة بنتائج **العمليات الحسابية إلى المستخدم . وتستخدم** تعليمات الإختبار لأختبار قيم خانات **كلمة أو حالة/شكل ناتج عملية حسابية ما ،** ومن ثم يتم إستخدام تعليمة تفرع **للإنتقال لمجموعة مختلفة من التعليمات تبعاً** للقرار الذى تم اتخاذه اعتماداً **على قيمة الخانة أو الحالة الناتجة من عملية حسابية**

5.1.4 عدد المعاملات

ما هو الحد الأقصى لعدد المعاملات المحتاجة لها التعليمة ؟ فمثلا ، التعليمات **الحسابية والمنطقية تتطلب معاملات كثيرة** ، وتقريبا كل العمليات الحسابية **والعمليات المنطقية هي إما أحادية** (معامل مصدر واحد) أو ثنائية (2 معامل **المصدر**) ، وبالتالي ، فإننا نحتاج كحد أقصى لعنوانين كمؤشرين لمعاملات **المصدر** (الداخلة فى العملية) . وناتج العملية يتم تخزينه فى عنوان ثالث يطلق **عليه معامل النتيجة/الوجهة** ، وأخيرا ، بعد الأنتهاء من التعليمة لا بد من جلب **التعليمة التالية لذلك كانت هناك حاجة لعنوانها**.

الشكل (5.3) **يقارن نماذج التعليمات بواحد** ، واثنين ، وثلاثة معاملات والتى **يمكن أن تستخدم لحساب المعادلة التالية** :

$$Y = (A - B) / [C + (D \times E)] .$$

باستخدام تعليمات ثلاثية كل تعليمة تحدد موقعين لمعاملين المصدر و آخر **لمعامل الناتج** ، ولأننا لا نريد أن نغير من قيمة أي من مواقع المعاملات ، لذلك **يجب أستخدام موقع مؤقت** س لتخزين بعض النتائج المؤقتة. وشكل التعليمة ذات **الثلاثة عناوين** ليس شائعاً وذلك لأنها تتطلب شكلاً طويلاً نسبياً بحيث تحمل **مؤشرات لثلاثة عناوين** .

وفى التعليمات بعنوانين يوجد عنوان واحد يجب أن يكون مزدوج الوظيفة **كمعامل مصدر وناتج** ، ولذلك ، فإن تعليمة SUB Y , B ، تنفذ إجراءات **حساب Y - B وتخزن النتيجة فى Y** . والتنسيق ثنائي العنوان يقلل من متطلبات

الحجم ، ولتجنب تغيير قيم المعامل المزدوج يتم استخدام تعليمة (MOVE) **لنقل** القيم لموقع مؤقت قبل تنفيذ العملية .

التعليمة	التعليق
SUB Y, A, B	Y ← A − B
MPY T, D, E	T ← D × E
ADD T, T, C	T ← T + C
DIV Y, Y, T	Y ← Y ÷ T

(أ) تعليمات بثلاث عناوين

التعليمة	التعليق
MOVE Y, A	Y ← A
SUB Y, B	Y ← Y − B
MOVE T, D	T ← D
MPY T, E	T ← T × E
ADD T, C	T ← T + C
DIV Y, T	Y ← Y ÷ T

(ب) تعليمات بعنوانين

التعليمة	التعليق
LOAD D	AC ← D
MPY E	AC ← AC × E
ADD C	AC ← AC + C
STOR Y	Y ← AC
LOAD A	AC ← A
SUB B	AC ← AC − B
DIV Y	AC ← AC ÷ Y
STOR Y	Y ← AC

(ج) تعليمات بعنوان واحد

الشكل (5.3) – ثلاث برامج لتنفيذ المعادلة $Y = (A - B) / [C + (D \times E)]$

أبسط شكل للتعليمة هو التعليمة بعنوان واحد (معامل واحد) . **ولكى تعمل التعليمة** المعامل الثاني يجب أن يكون ضمنيا ، وهذا كان شائعا **في المعالجات المبكرة،** فالعنوان الضمني هو لمسجل فى المعالج يعرف باسم المجمع (AC) . **ويحتوي** المجمع على أحد المعاملات وكذلك يستخدم لتخزين النتيجة ، **وفي مثالنا السابق** تم إنجاز المعادلة فى ثمانية تعليمات.

بعض التعليمات يمكن أن تنفذ بدون عنوان (بدون معامل) ، **التعليمات التي بدون** عنوان (صفر) **تطبق على تنظيم معين للذاكرة يسمى بالمكدس . المكدس يعمل** بأسلوب الداخل أخيراً – الخارج اولاً ، **فعنوان المكدس معروف حيث الأشارة** دائما الى قمة المكدس (**عنوان الموقع الذى فى القمة) ويخصص مسجل خاص**

فى المعالج لحفظ عنوان القمة ، ولذلك التعليمات بصفر عنوان تشير وتتعامل **دائما مع العناصر التى بقمة المكدس** .

الجدول (5.1) **يلخص** أوضاع التعليمات من الصفر أو واحد أو اثنين أو ثلاثة **عناوين . وفي كل حالة** من المفترض أن عنوان التعليمة التالية مشار اليها ضمنياً، **ولذلك العملية يتم** اجراءها بمعاملين معامل مصدر ومعامل ناتج .

عدد العناوين لكل تعليمة هو قرار أساسي فى تصميم المعالج ، فعدد أقل من **العناوين لكل تعليمة** ينتج عنه تعليمات بدائية جداً ، ومعالج أقل تعقيداً ، وتعليمات **أقل طولاً ، ومع أزدياد** عدد التعليمات فى البرنامج فأن ذلك يزيد من زمن التنفيذ **و بالتالى برنامج أطول** و أكثر تعقيداً. هناك مفارقة هامة مابين العنوان الواحد و **عدة عناوين ، ففى العنوان** الواحد للمبرمج مسجل واحد متعدد الأغراض **(المجمع)** مما يفرض التعامل مع الذاكرة الرئيسية ، بينما فى التعليمات المتعددة **العناوين يوجد** عدة مسجلات متعددة الأغراض و هذا يؤدى إلى سرعة فى التنفيذ **نتيجة أن الأشارة** الى المسجل أسرع منه الى الذاكرة . وأغلب المعالجات تستخدم **خليط من التعليمات** الثنائية والثلاثية العنوان (المعامل) وذلك بسبب المرونة **والقدرة على** أستخدام عدة مسجلات .

الجدول (5.1) –أشكال التعليمة بإختلاف عدد العناوين (بدون تعليمات تفرع)

عدد العناوين	التمثيل الرمزي	التفسير
3	OP A, B, C	A ← B OP C
2	OP A, B	A ← A OP B
1	OP A	AC ← AC OP A
0	OP	T ← (T – 1) OP T

AC = المركم
T = أعلى المكدس
(T – 1) = العنصر الثاني للمكدس
A, B, C = مواقع ذاكرة أو مسجل

5.2 أنواع المعاملات

تشتغل تعليمات المعالج على البيانات وأكثر الأصناف العامة للبيانات أهمية هي:

1. العناوين : العناوين هي - في الواقع - شكل من أشكال البيانات . وفي كثير من الحالات يجب إجراء بعض العمليات الحسابية على مؤشر المعامل في التعليمة لتحديد عنوان فى الذاكرة الرئيسية أو الظاهرية ، وفي هذا السياق يمكن إعتبار العناوين على أنها أعداد صحيحة بدون اشارة .
2. الأعداد : هناك ثلاثة أنواع شائعة من البيانات الرقمية في أجهزة الحاسب :
   - عدد ثنائي صحيح أو ثنائي بنقطة ثابتة .
   - ثنائي بالنقطة العائمة .
   - عشري .
3. الأحرف : النموذج الشائع للبيانات هو النص أو سلسلة أحرف. أن البيانات النصية هي الأكثر ملائمة للإنسان ، ولكن لا يمكن تخزينها بصيغتها الحرفية بسهولة أو نقلها بواسطة أنظمة معالجة البيانات والأتصالات ، والأنظمة الحاسوبية مصممة للعمل على البيانات الثنائية ، لهذا فقد وضعت عددا من الرموز التي تمثل الأحرف كنسق من الثنائيات . والمثال الشائع على ذلك هو نظام الترميز (ASCII) .
4. البيانات المنطقية : عادة ما يتم التعامل مع الكلمة الثنائية كوحدة واحدة من البيانات الثنائية ، ولكن من المفيد أحيانا النظر إلى وحدة **تتكون من** *ن* **من** الخانات على أنها تتكون من *ن* من الوحدات الثنائية المنفردة ، **وكل عنصر موجود** ينظر إليه أما 0 أو 1 ، **وبهذه الطريقة يتم التعبير عنها كبيانات**

منطقية ويتم التعامل مع كل خانة ثنائية بإنفراد و إجراء أى عمليات منطقية عليها (عمليات الحساب المنطقى الثنائى) .

5.3 أنواع العمليات

عدد رموز العمليات يختلف من معالج إلى آخر ، ومع ذلك فانها تشترك فى نفس **الأنواع العامة للعمليات** ، وهذه الانواع هي:

- نقل البيانات .
- الحسابية .
- المنطقية .
- التحويل .
- الإدخال/الإخراج .
- التحكم بالنظام .
- نقل السيطرة .

والجدول (5.2) يسرد بعض التعليمات لكل نوع .

هذا القسم يقدم مناقشة موجزة للإجراءات التى يتخذها المعالج لتنفيذ كل نوع من العمليات (الاجراءات موجزة في الجدول (5.3)) .

نقل البيانات

النوع الأساسى من تعليمات المعالج هي تعليمات نقل البيانات . فيجب على **تعليمات نقل البيانات تحديد عدة أشياء** ، أولا ، يجب تحديد موقع المعامل المصدر **والوجهة ، والموقع يمكن أن يكون فى الذاكرة أو المسجل أو قمة المكدس** . ثانيا، **لابد من الإشارة إلى طول البيانات التي سيتم نقلها** . ثالثا ، وكما هو الحال مع

جميع التعليمات بالمعاملات يجب تحديد طريقة العنونة لكل معامل (آلية حساب عنوان المعامل) .

الجدول (5.2) – العمليات الشائعة فى فئات التعليمات (1/3)

النوع	أسم العملية	وصف العملية
نقل البيانات	Move (transfer)	نقل كلمة أو قالب من مصدر الى وجهة
	Store	نقل كلمة من المعالج الى الذاكرة
	Load (fetch)	نقل كلمة من الذاكرة الى المعالج
	Exchange	أستبدال محتويات مصدر مع وجهة
	Clear (reset)	نقل كلمة مكونة من أصفار (0...0) الى وجهة
	Set	نقل كلمة مكونة من آحاد (1...1) إلى وجهة
	Push	نقل كلمة من مصدر إلى قمة المكدس
	Pop	نقل كلمة من قمة المكدس الى وجهة
التحويل	Translate	ترجمة قيم فى مقطع من الذاكرة بناء على جدول مطابقة
	Convert	تحويل محتوى كلمة من شكل الى آخر (عشرى الى ثنائى)
الحسابية	Add	حساب مجموع معاملان
	Subtract	حساب الفرق بين معاملان
	Multiply	حساب ضرب معاملان
	Divide	حساب قسمة معاملان
	Absolute	أستبدال معامل بقيمته المطلقة
	Negate	تغيير إشارة معامل
	Increment	زيادة 1 الى معامل
	Decrement	انقاص 1 من معامل

الجدول (5.2) – العمليات الشائعة فى فئات التعليمات (2/3)

النوع	أسم العملية	وصف العملية
المنطقية	AND	إجراء عملية (AND) منطقية
	OR	إجراء عملية (OR) منطقية
	NOT (Complement)	إجراء عملية (NOT) منطقية
	EXOR	إجراء عملية (EXOR) منطقية
	Test	إختبار شرط محدد: رفع (تصبح قيمتها =1) الأعلام بناء على نتيجة عملية تمت فى وحدة الحساب و المنطق
	Compare	إجراء مقارنة منطقية او حسابية لمعاملان او اكثر ، تعريف (ترفع) قيم الأعلام بناء على الناتج
	Set Control Variables	صنف من التعليمات لتهيئة التحكم فى روتين المقاطعة (تقديم الخدمة لمقاطعة) ، التحكم فى ساعة النظام ، أو التحكم فى اجراءات حماية ، أو غيره
	Shift	إزاحة معامل الى اليمين (أو اليسار) مع إضافة ثابت فى النهاية
	Rotate	إزاحة معامل الى اليمين (أو اليسار) مع لف النهاية (حلقة)
الإدخال/الإخراج	Input (read)	نقل بيانات من منفذ أو جهاز أدخال/إخراج محدد إلى وجهة (ذاكرة أو مسجل بالمعالج)
	Output (write)	نقل بيانات من مصدر محدد إلى منفذ أو جهاز أدخال/إخراج
	Start I/O	نقل تعليمات الى معالج الإدخال/الإخراج لبدء عملية أدخال/إخراج
	Test I/O	نقل معلومات الحالة من نظام الإدخال/الإخراج الى وجهة محددة

الجدول (5.2) – العمليات الشائعة فى فئات التعليمات (3/3)

النوع	اسم العملية	وصف العملية
نقل السيطرة	Jump (branch)	أنتقال غير مشروط ، تحميل عداد البرنامج بعنوان محدد
	Jump Conditional	اختبار شرط معين، أما تحميل عداد البرنامج بعنوان محدد أو لا تعمل شئ ، حسب الشرط
	Jump to Subroutine	حفظ المحتويات الحالية لعداد البرنامج و غيرها من مسجلات فى موقع معروف، الأنتقال الى عنوان محدد
	Return	إستبدال محتوى عداد البرنامج و المسجلات الأخرى من موقع معروف
	Execute	جلب معامل من موقع محدد و التنفيذ حسب التعليمة ، لا تعديل لعداد البرنامج
	Skip	زيادة عداد البرنامج لتخطى التعليمة التالية
	Skip Conditional	أختبار شرط معين، أما تخطى أو لا تعمل شئ ، حسب الشرط
	Halt	إيقاف تنفيذ برنامج
	Wait (hold)	إيقاف تنفيذ برنامج ، تكرار اختبار شرط ، إستئناف التنفيذ حال تحقق الشرط
	No Operation	لا تجرى أى عملية لكن تنفيذ البرنامج يستمر

الحسابية

معظم المعالجات توفر العمليات الحسابية الأساسية كالجمع والطرح والضرب والقسمة للأعداد الصحيحة بالأشارة (ذات نقطة ثابتة) ، وغالبا ما يتم توفيرها

أيضا للأعداد العشرية ذات النقطة العائمة ، وتوجد عمليات أخرى متنوعة محتملة ضمن هذا النوع من العمليات وتشمل على سبيل المثال :

- **القيمة المطلقة** : أخذ القيمة المطلقة للمعامل .
- **النفي** : المعكوس المنطقي للمعامل .
- **الزيادة** : إضافة 1 إلى المعامل .
- **النقصان** : طرح 1 من المعامل .

المنطقية

معظم المعالجات توفر مجموعة متنوعة من التعليمات لمعالجة خانات الكلمة كل على حدة وإجراء عمليات منطقية عليها . وبعض العمليات المنطقية الأساسية التي يمكن القيام بها على البيانات الثنائية هى : عملية (NOT) أو (AND) أو (OR) أو (XOR) وهي أكثر العمليات المنطقية بمعاملين شيوعا . بالإضافة إلى العمليات المنطقية المختصة بالمنطق الثنائى فأن معظم المعالجات توفر مجموعة متنوعة من تعليمات الأزاحة والتدوير . والشكل (5.4) يوضح هذه العمليات الأساسية ، ففى الإزاحة المنطقية تزاح أجزاء الكلمة من اليسار أو اليمين وتفقد خانة واحدة فى النهاية ، وفى التدوير أو الإزاحة الدائرية العملية تحافظ على كافة الخانات الخاضعة للعملية .

التحويل

تعليمات التحويل هي تلك التي يمكنها تغيير تنسيق البيانات أو تعمل على تنسيق البيانات ، فعلى سبيل المثال التحويل من النظام العشري إلى النظام الثنائي .

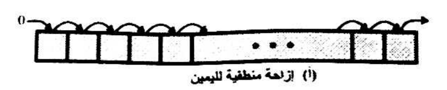
الإدخال / الإخراج

هناك مجموعة متنوعة من الأساليب لتقنيات الإدخال/الإخراج منها الإدخال/الإخراج المبرمج ، والإدخال/الإخراج المبرمج باستخدام الذاكرة ،

والوصول المباشر للذاكرة (DMA) أو إستخدام معالج خاص بالإدخال/الإخراج، وأكثر المعالجات لا توفر سوى عدد قليل من التعليمات الخاصة بالإدخال/الإخراج .

الجدول (5.3) – عمل المعالج فى كل نوع من العمليات

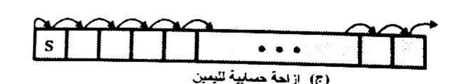
نقل البيانات	نقل البيانات من موقع الى آخر.
	أذا تضمنت التعامل مع الذاكرة: ▪ تحديد العنوان بالذاكرة. ▪ إجراء التحويل من عنوان ظاهرى الى حقيقى. ▪ أختبار الذاكرة السريعة. ▪ بدء القراءة/الكتابة من الذاكرة.
الحسابية	ربما تتضمن نقل بيانات قبل أو/و بعد
	إجراء عملية وظيفية فى وحدة الحساب والمنطق .
	تعريف الأعلام (رفع قيمتها الى 1).
المنطقية	مثل الحسابية.
التحويل	مثل الحسابية و المنطقية ، ربما تتضمن منطق خاص لإجراء التحويل.
نقل السيطرة	تعديل عداد البرنامج نتيجة الأستدعاء/العودة من روتين جزئى وتدير عملية تمرير المعاملات و الربط بينهما.
الإدخال/الإخراج	تصدر أوامر الى وحدة الإدخال/الإخراج.
	إذا كان الإدخال/الإخراج مقترن بالذاكرة ، تحدد العنوان المناظر بالذاكرة.



(أ) إزاحة منطقية لليمين



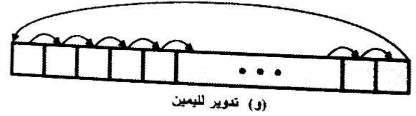
(ب) إزاحة منطقية لليسار



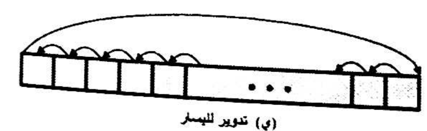
(ج) إزاحة حسابية لليمين



(د) إزاحة حسابية لليسار



(و) تدوير لليمين



(ي) تدوير لليسار

الشكل (5.4) – عملية الإزاحة والتدوير

التحكم بالنظام

تعليمات التحكم بالنظام هي تلك التي يمكن تنفيذها فقط عندما يكون المعالج في حالة معينة وبصلاحيات خاصة أو تنفذ برنامجا في منطقة مميزة وخاصة من الذاكرة ، وعادة ما يتم تنفيذ هذه التعليمات من قبل نظام التشغيل .

نقل السيطرة

في جميع أنواع العمليات السابقة فأن التعليمة التالية التي يتعين القيام بها بعد التعليمة الحالية هي التي تليها مباشرة في الذاكرة . ومع ذلك ، فإن جزءاً كبيراً من التعليمات في أي برنامج وظيفتها تغيير تسلسل تنفيذ التعليمات ، ومن أجل هذه التعليمات يقوم المعالج بعملية تحديث لعداد البرنامج إلى عنوان تعليمة ما في الذاكرة .

ننتقل الآن إلى مناقشة عمليات نقل السيطرة الأكثر شيوعا والموجودة في طقم التعليمات : التفرع والقفز واستدعاء الإجراء.

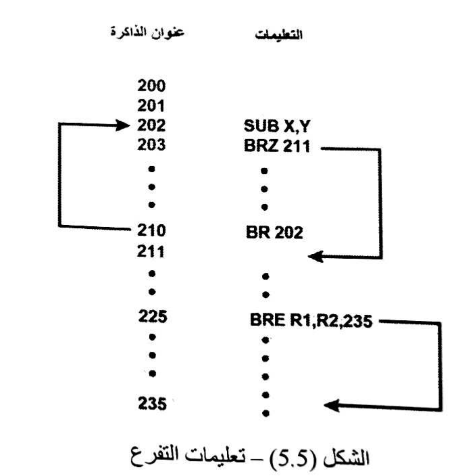
تعليمات التفرع : أحد معاملات تعليمة التفرع هو عنوان التعليمة التالية وذلك لجلبها وتنفيذها ، وهي في معظم الأحيان تعليمة تفرع مشروط ، بمعنى أن التفرع يتم إذا تحقق شرط معين وإلا يتم تنفيذ التعليمة التالية في التسلسل (زيادة عداد البرنامج كالعادة) . وتعليمة التفرع التي ينفذ فيها التفرع دائما هي تفرع دون قيود أو شروط.

وهناك طريقتان لإختبار حالة المعالج و ذلك لإجراء/تنفيذ تعليمة تفرع مشروط.
أولا ، أختبار خانة بمسجل الحالات الذي يعكس حالة ناتج عملية حسابية ومنطقية، ويمكن أن يكون هناك أربعة أنواع مختلفة من تعليمات التفرع المشروط :

BRP X	تفرع للموقع X إذا كانت النتيجة إيجابية
BRN X	تفرع للموقع X إذا كانت النتيجة سلبية
BRZ X	تفرع للموقع X إذا كانت النتيجة صفر
BRO X	تفرع للموقع X إذا حدث فيض

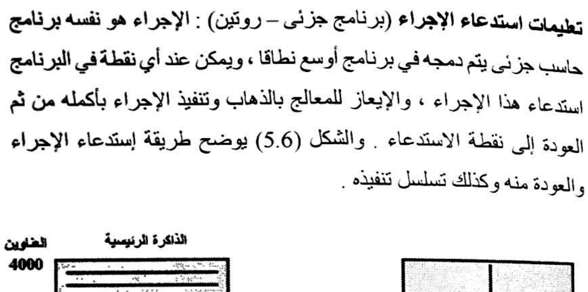
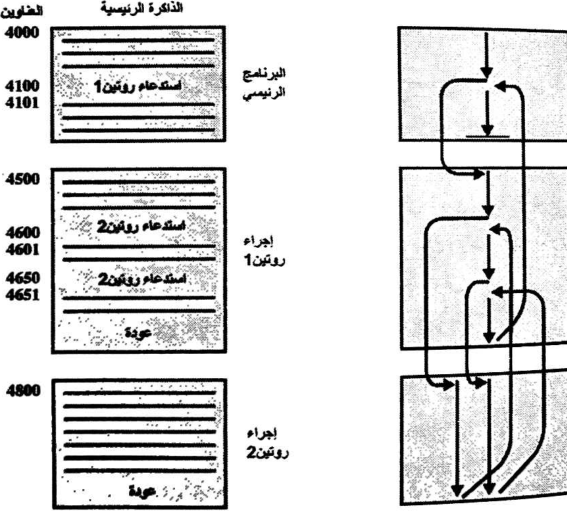
ففي التعليمات السابقة فإن نتيجة العملية الأخيرة التى تم تنفيذها فى وحدة الحساب والمنطق هى التى تُعين (ترفع) علم الحالة (حالة الناتج : أيجابية أو سلبية أو صفر أو فيض) وجرى إختباره بواسطة هذه التعليمات . وفي الأسلوب الأخر يمكن إستخدام شكل من التعليمات بثلاثة عنوانين وذلك لإجراء المقارنة وتحديد التفرع في نفس التعليمة . والشكل (5.5) يوضح آلية عمل بعض أنواع التفرعات المذكورة سالفاً.

تعليمات التخطى : شكل آخر من أشكال نقل السيطرة هى تعليمة التخطى . تعليمات التخطي تتضمن عنوانا ضمنياً ، والتخطي عادة يعني تخطي تعليمة واحدة ، وبالتالي فإن العنوان الضمني يساوي عنوان التعليمة التالية مضاف أليه طول التعليمة.



الشكل (5.5) – تعليمات التفرع

تعليمات استدعاء الإجراء (برنامج جزئى – روتين) : **الإجراء هو نفسه برنامج** حاسب جزئى يتم دمجه في برنامج أوسع نطاقا ، ويمكن عند أي نقطة في البرنامج استدعاء هذا الإجراء ، والإيعاز للمعالج بالذهاب وتنفيذ الإجراء بأكمله من ثم العودة إلى نقطة الاستدعاء . والشكل (5.6) يوضح طريقة إستدعاء الإجراء والعودة منه وكذلك تسلسل تنفيذه .

الشكل (5.6) – الأجراءات المتداخلة

5.4 أساليب العنونة و تنسيقات التعليمة

فى الاجزاء السابقة جرى التركيز على ماذا تنفذ أو تفعل التعليمات ، وبشكل خاص تَم تناول أنواع المعاملات و العمليات التى تحددها تعليمات المعالج . وفى هذا الجزء سنتناول كيفية تحديد المعاملات والعمليات التى تنفذها التعليمات ، وهنا تبرز قضيتين ، الأولى ، كيفية تحديد عنوان المعامل ، والثانية ، كيفية تنظيم خانات التعليمة لكى تُعرِّف عنوان المعامل و العملية المطلوب إجراءها .

5.4.1 أساليب العنونة

حقل أو حقول العنوان في الصيغة النموذجية للتعليمة صغير نسبيا . وفى العادة نود أن نكون قادرين على الإشارة إلى مجموعة كبيرة من المواقع في الذاكرة الرئيسية ، أو بالنسبة لبعض النظم ، الذاكرة الظاهرية . ولتحقيق هذا الهدف أستخدمت مجموعة متنوعة من أساليب العنونة وهى تنطوي على نوع من المفاضلة مابين مدى العنوان و/أو مرونة العنونة من جهة ، ومن جهة أخرى ، عدد المؤشرات للذاكرة في التعليمة و/أو تعقيد حساب العنوان .

وفي هذا الجزء ندرس أساليب العنونة الأكثر شيوعا ، وهى :

- **الفورية .**
- **المباشرة .**
- **غير المباشرة .**
- **المسجل .**
- **المسجل غير المباشرة .**
- **الإزاحة .**
- **المكدس .**

وتتضح أوضاع هذه الأساليب في الشكل (5.7) ، **وفي هذه الجزء سوف نستخدم** الأختصارات التالية :

(س) = محتويات موقع الذاكرة س أو المسجل س .

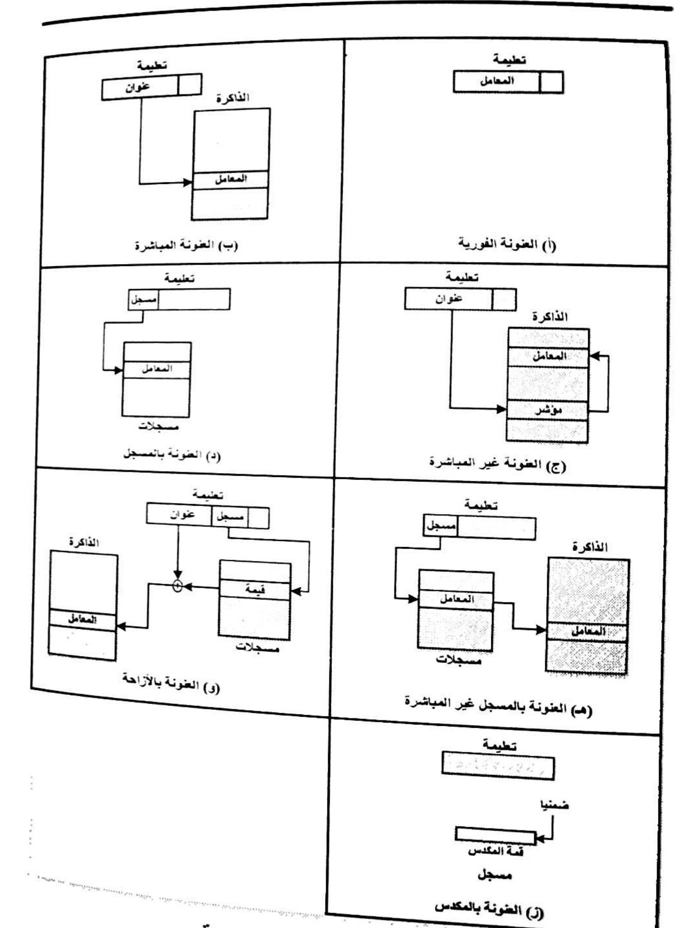
ع ف = العنوان الفعلي (المؤثر) للموقع الذي يحتوي على المعامل المؤشر أليه .

م = محتويات حقل العنوان في التعليمة التى تشير إلى مسجل .

ع = محتويات حقل العنوان في التعليمة .

الجدول (5.4) يوضح كيفية إجراء حساب العنوان لكل أسلوب عنونة ، **والجدير** بالذكر هنا إن بنية المعالجات توفر أكثر من اسلوب للعنونة **بحيث يحدد رمز** العملية نوع العنونة المستخدمة بالتعليمة . **والملاحظة الثانية تتعلق بتفسير العنوان** الفعلى (ع ف) .

ففي الأنظمة التى بدون ذاكرة ظاهرية ، فإن **العنوان الفعلى يكون إما عنوان** بالذاكرة الرئيسية أو لمسجل ، ولكن في **أنظمة الذاكرة الظاهرية ، العنوان الفعلي** هو عنوان ظاهري أو لمسجل ، **والمطابقة أو التغيير الفعلى إلى العنوان** المادى/الحقيقى هو وظيفة وحدة إدارة الذاكرة (MMU) **وغير مرئي للمبرمج** أو المستخدم .



الشكل (5.7) – أساليب العنونة

الجدول (5.4) – أساليب العنونة الأساسية

أسلوب العنونة	الخوارزمية	الميزة	العيب
الفورية	المعامل = ع	لا توجد أشارة للذاكرة	مقدار محدد للمعامل
المباشرة	ع ف = ع	بسيط	سعة محدودة للذاكرة
غير المباشرة	ع ف = (ع)	سعة كبيرة للذاكرة	تأشير متعدد للذاكرة
المسجل	ع ف = م	لا توجد أشارة للذاكرة	سعة محدودة للذاكرة
المسجل غير المباشرة	ع ف = (م)	سعة كبيرة للذاكرة	تأشير اضافى للذاكرة
الأزاحة	ع ف = ع + (م)	مرنة	التعقيد
المكدس	ع ف = قمة المكدس	لا توجد إشارة للذاكرة	تطبيق محدود

العنونة الفورية

أبسط أسلوب من أساليب العنونة هى العنونة الفورية حيث قيمة المعامل موجودة في التعليمة (المعامل = ع) ، ويمكن إستخدام هذا الوضع لتقديم وإستخدام الثوابت أو تعريف قيم أولية للمتغيرات . ومن مميزات العنونة الفورية أنه لا حاجة للإشارة للذاكرة الرئيسية إلا لجلب التعليمة للحصول على المعامل وبذلك نوفر من زمن التنفيذ . والعيب هو أن حجم المعامل محدود بحجم حقل العنوان والذى في معظم مجموعات التعليمات صغير مقارنة مع طول الكلمة الثنائية.

العنونة المباشرة

شكل بسيط آخر للعنونة هو العنونة المباشرة ، وفيها حقل العنوان بالتعليمة يحتوي على العنوان الفعلى للمعامل : ع ف = ع . وهذه التقنية شائعة في الأجيال السابقة

للمعالجات ولكن ليست شائعة في الأبنية المعاصرة للمعالجات وهى تتطلب إشارة **واحدة فقط للذاكرة** بدون أى حسابات خاصة . وهذه التقنية توفر فضاء محدودة **للعناوين (أقصى سعة** قابلة للعنونة من الذاكرة محدودة بطول حقل العنوان **بالتعليمة) .**

العنونة غير المباشرة

أحد الحلول للتغلب على عيوب العنونة المباشرة هو أن يكون حقل العنوان يشير **إلى عنوان كلمة في** الذاكرة ، والذي بدوره يحتوي على العنوان الكامل للمعامل **ويعرف هذا** بإسم العنونة غير المباشرة : ع ف = (ع) . والميزة الواضحة لهذا **النهج هى لو** أن طول الكلمة ن – خانة فإن مساحة العناوين المتوفرة الأن هى $2^{ن}$. **والعيب هو أن تنفيذ التعليمة** يتطلب زمناً أطول بحيث يتطلب مراجعة الذاكرة **مرتين لجلب المعامل** : واحدة للحصول على العنوان والثانية للحصول على قيمته. **ونادراً ما يتم** إستخدام العنونة غير المباشرة المتعددة المستويات أو المتتالية حيث ع ف = (...(ع)...) .

العنونة بالمسجل

العنونة بالمسجل شبيهة بالعنونة المباشرة ، والفرق الوحيد هو أن حقل العنوان **بالتعليمة يشير إلى مسجل** حيث يوجد المعامل بدلاً من عنوان بالذاكرة الرئيسية : **ع ف = م** . وللتوضيح ، إذا كانت محتويات حقل عنوان المسجل في التعليمة هى 5 ، أذن المسجل م5 هو المقصود ، وقيمة المعامل موجودة فى م5 . نموذجياً ، **حقل العنوان الذى يشير** للمسجلات مكون من 3 الى 5 خانات (ثنائى) ، وبذلك **يكون إجمالى المسجلات** التى يمكن التعامل معها والإشارة اليها هى من 8 الى ... مسجلاً للإغراض العامة .

مزايا عنونة المسجل (1) أن حقل العنوان صغير في التعليمة ، (2) والإشارة إليه لا تستهلك وقتا حيث أن التواصل مع مسجل داخل المعالج يأخذ وقتاً أقصر بكثير منه لعنوان موقع بالذاكرة الرئيسية . وعيب العنونة بالمسجل أن مساحة العنوان محدودة جداً . فإذا تم أستعمال أسلوب العنونة بالمسجل بكثافة فى طقم التعليمات ، فهذا يعنى أن مسجلات المعالج سوف تستخدم بكثافة ونتيجة لأن عدد المسجلات محدود جداً (بالنسبة لعدد المواقع فى الذاكرة الرئيسية) فإن إستخدامها بهذه الطريقة له مردود مباشر إذا استعملت بكفاءة ، فإذا كان كل معامل يتم جلبه من الذاكرة الرئيسية لمسجل ما يتم إستخدامه لمرة واحدة من ثم يعاد للذاكرة ، فإن هذا الإجراء يستنزف الزمن وإجراءات التحكم لأنه عبارة عن إضافة خطوة وسيطة ، وبالمقابل ، إذا بقى المعامل فى المسجل لعدة إستخدامات ثم يعاد إلى الذاكرة بهذا تتم الإستفادة القصوى منه ونتحصل على كفاءة حقيقية من إستخدام المسجلات وذلك بتوفير فى زمن جلب المعاملات والإجراءات التى تستلزم ذلك.

العنونة غير المباشرة بالمسجل

العنونة غير المباشرة بالمسجل مناظرة للعنونة غير المباشرة ، حيث أن فى العنونة غير المباشرة بالمسجل حقل العنوان بالتعليمة يشير إلى مسجل (يوجد به عنوان مسجل) الذى بدوره يحتوى على عنوان بالذاكرة للموقع الذى يحتوى على المعامل ، وبأختصار ، ع ف = (م) . ومزايا وعيوب العنونة غير المباشرة بالمسجل نفسها للعنونة غير المباشرة . ففي كلتا الحالتين يتم التغلب على قيود مساحة العنونة (نطاق محدود من العناوين) فى حقل العنوان من خلال أن الموجود فى هذا الحقل يشير إلى موقع بعرض كلمة يحتوي على العنوان ،

وبالإضافة إلى ذلك ، فالعنونة غير المباشرة بالمسجل تستخدم الإشارة للذاكرة أقل منه في العنونة غير المباشرة .

العنونة بالإزاحة

هناك طريقة فعالة جداً للعنونة تجمع بين قدرات العنونة المباشرة والعنونة غير المباشرة بالمسجل وسوف نشير إليها بالعنونة بالإزاحة : ع ف = ع + (م) . **العنونة بالإزاحة تتطلب أن يكون بالتعليمة حقلي عنوان ،** إحدها يجب أن يكون **صريحاً (قيمته = ع ، فرضاً)** ويتم استخدامه مباشرة ، والقيمة الاخرى الواردة **في الحقل الآخر تشير (مؤشر)** إلى مسجل بحيث تتم إضافة محتوياته إلى ع **لإنتاج العنوان الفعلي .**

العنونة بالإزاحة لها عدة صور ، منها :

العنونة النسبية : النسبية هنا هى ضمنياً إلى القيمة التي بمسجل عداد **البرنامج ، بحيث** أن عنوان التعليمة التالية (قيمة مسجل عداد البرنامج) **يضاف لمحتويات** حقل العنوان لإنتاج العنوان الفعلى (ع ف) . العنوان **الفعلى هو** إزاحة نسبية من عنوان التعليمة الحالية (قيمة مسجل عداد **البرنامج) .**

الفهرسة : وتعنى أن حقل العنوان يؤشر لعنوان بالذاكرة الرئيسية ، **والمسجل المشار إليه يحتوي** على مقدار الإزاحة الموجبة لذلك العنوان. **إذاً العنوان الفعلى هو العنوان المشار إليه** فى التعليمة مضاف إليه مقدار **الإزاحة الموجودة بالمسجل ،** ع ف = ع + (م) . المسجل المستعمل فى **هذا الأسلوب من العنونة ممكن أن يكون مسجل خاص بالفهرسة أو مسجل متعدد الأغراض (عام).**

العنونة بالمكدس

المكدس هو صف خطي من المواقع و يشار إليها أحيانا بأسم **طابور آخر-دخول** اول-خروج (LIFO) حيث أن العناصر يتم أضافتها **إلى أعلى أو قمة المكدس.** ويرتبط بالمكدس مؤشر قيمته تؤشر دائما إلى عنوان قمة المكدس ، **ويتم الحفاظ** على قيمة مؤشر المكدس في مسجل خاص ، وبالتالي ، **فإن الإشارة إلى موقع** بالمكدس في الذاكرة هي في الحقيقة عنونة غير مباشرة بالمسجل .

5.4.2 أساليب العنونة لمعالجات آنتل (Intel)

الجدول (5.5) يوضح أوضاع العنونة المستخدمة فى المعالجات آنتل x86 **. ففى** العنونة الفورية يتم تضمين المعامل في التعليمة ، **وبهذا يمكن أن يكون المعامل** 8-خانات أو كلمة أو كلمة مزدوجة من البيانات . **وفى حالة المعامل بالمسجل ،** فإن قيمة المعامل توجد في المسجل . للتعليمات العامة **مثل نقل البيانات ،** والحسابية ، والتعليمات المنطقية ، فيمكن أن يكون المعامل **فى أحد المسجلات** العامة ذات 32-خانة ، أو أحد المسجلات العامة ذات 16-خانة ، **أو أحد** المسجلات العامة ذات 8-خانات ، وهناك أيضا **بعض التعليمات التي تشير إلى** مسجلات محدد المقطع (جزء معين من الذاكرة الرئيسية).

بقية أوضاع أو أساليب العنونة تؤشر لمواقع في الذاكرة . **ويجب تحديد موقع** الذاكرة بالمقطع الذي يحتوي على الموقع و مقدار **الإزاحة عن بداية هذا المقطع.** في بعض الحالات تم تحديد المقطع بشكل صريح ، **وفي حالات أخرى يتم تحديد** المقطع **من خلال قواعد بسيطة تحدد المقطع .**

الجدول (5.5) - صيغ العنونة للمعالجات أنتل x86

اسلوب العنونة	الخوارزمية
الفورية	A = المعامل
معامل المسجل	LA = R
الإزاحة	LA = (SR) + A
القاعدة	LA = (SR) + (B)
القاعدة بالإزاحة	LA = (SR) + (B) + A
فهرسة بمقياس والإزاحة	LA = (SR) + (I) × S + A
القاعدة بالفهرسة والإزاحة	LA = (SR) + (B) + (I) + A
القاعدة بالفهرسة بمقياس والإزاحة	LA = (SR) + (I) × S + (B) + A
النسبية	LA = (PC) + A

R = مسجل
B = مسجل القاعدة
I = مسجل الفهرسة
S = عامل المقياس
LA = العنوان الخطى
(x) = قيمة x
SR = مسجل المقطع
PC = عداد البرنامج
A = محتوى حقل العنوان بالتعليمة

في وضع الإزاحة (العنونة بالإزاحة) فإن مقدار إزاحة المعامل موجود كجزء من التعليمة بطول 8 او 16 أو 32 خانة للإزاحة . وفى المقاطع فإن جميع العناوين في التعليمات تشير إلى إزاحة داخل المقطع .

بقية طرق العنونة غير مباشرة بحيث أن الجزء الخاص بالعنوان فى التعليمة يخبر المعالج بمكان تواجد العنوان .

5.4.3 تنسيق التعليمة

شكل أو تنسيق التعليمة يُحدد وضع خانات التعليمة و بماذا تختص . **فشكل التعليمة** يجب أن يتضمن رمز العملية و - ضمناً أو صراحةً - صفر أو أكثر من المعاملات. وتتم الإشارة إلي كل معامل صريح بإستخدام إحدى وسائل العنونة المذكورة سلفاً . فالتنسيق أو شكل التعليمة يجب (ضمناً أو صراحةً) أن يشير إلى وسيلة العنونة لكل معامل ، وفى أغلب أطقم التعليمات يتم إستخدام أكثر من تنسيق للتعليمة .

تصميم نسق التعليمة هو فن معقد وتوجد مجموعة متنوعة ومذهلة من التصميمات لهذا الغرض ، وفى التالى نتناول المسائل الأساسية فى التصميم مع توضيح لبعض نقاطه .

طول التعليمة

أكثر المسائل أهمية فى أسس تصميم معمارية طقم التعليمات والتى ينبغي مواجهتها هى طول نسق التعليمة . فهذا القرار يؤثر، ويتأثر بحجم الذاكرة وتنظيم الذاكرة وهيكلية الناقل وتعقيد المعالج وكذلك سرعة المعالج . إن هذا القرار يحدد ثراء ومرونة المعالج من وجهة نظر المبرمج بلغة التجميع **(لغة الآلة الأبجدية).** المفاضلة الأكثر وضوحا هنا هى بين **الرغبة في تعليمة قوية والحاجة لتوفير** المساحة ، فالمبرمجين يريدون المزيد من الرموز **لعمليات أكثر (تعليمات) ،** ومعاملات أكثر ، وأساليب عنونة أكثر ، وزيادة مدى العناوين . **فالمزيد من** رموز العمليات و معاملات أكثر **تساعد المبرمجين لأنه يُمكن من كتابة برامج** أقصر لإنجاز مهام معينة . **وبالمثل ، أساليب أو طرق عنونة أكثر تعطى المبرمج** قدراً أكبر من المرونة في تحقيق وظائف معينة ، **مثل التعامل بالجداول و التفرع**

متعدد الاتجاهات . وبطبيعة الحال مع الزيادة في حجم الذاكرة الرئيسية و إزدياد استخدام الذاكرة الظاهرية فإن المبرمجين يريدون أن تكون لهم القدرة للتواصل مع نطاق واسع من عناوين الذاكرة (زيادة سعة الذاكرة القابلة للعنونة) . إن كل هذه الأمور (رموز العمليات ، المعاملات ، وصيغ العنونة ، مدى العناوين) تتطلب خانات مخصصة لذلك مما يدفع في إتجاه تعليمة أطول ولكن تعليمة أطول قد يكون قراراً غير سليم ، فتعليمة 64-خانة تحتل مرتين مساحة تعليمة 32-خانة ولكن ربما قد تكون مفيدة أقل من مرتين .

وراء هذه المفاضلة الأساسية هناك إعتبارات أخرى ، إما ان يكون طول التعليمة مساويا لعرض ناقل الذاكرة (في نظام الناقل هو عرض ناقل البيانات) أو احدهما يجب أن يكون متكراراً للآخر وإلا فإننا لن نحصل على عدد لا يتجزأ من التعليمات خلال دورة الجلب . وهناك إعتبار آخر ذو صلة هو معدل نقل الذاكرة ، فبناء على ذلك يمكن للذاكرة أن تصبح عنق الزجاجة إذا كان المعالج يستطيع تنفيذ التعليمات بشكل أسرع مما يمكن أن تجلب له . أحد الحلول لهذه المشكلة هى إستخدام ذاكرة التخزين السريعة ، وآخر هو إستخدام تعليمات أقصر وبالتالي يمكن لتعليمة بطول 16-خانة ان تجلب بمعدل أسرع مرتين من تعليمة 32-خانة ولكن ربما يستغرق زمن تنفيذها أقل من نصف زمن تعليمة 32-خانة .

تخصيص خانات التعليمة

تخصيص الخانات فى نسق التعليمة هو مسألة صعبة وكذلك المفاضلة فيه معقدة. لتعليمة بطول معين هناك مفاضلة بين عدد رموز العمليات والقدرة على عنونة أكبر . فرموز لعمليات أكثر تعنى خانات أكثر في حقل رمز العملية وهذا يقلل من عدد الخانات المتاحة للعنونة . وهناك تحسين لهذه المفاضلة وهى إستخدام

رموز تعليمات متغيرة الطول ، وفى هذا الأسلوب يوجد حد أدنى لطول رمز التعليمة ولكن بعض رموز التعليمات يمكنها وصف عمليات إضافية بإستخدام خانات إضافية للتعليمة . بالنسبة للتعليمة الثابتة الطول ، فتوجد خانات قليلة للعنونة لذلك فهى تستعمل للتعليمات التى تتطلب معاملات قليلة و/أو أساليب عنونة محدودة .

إستخدام الخانات المخصصة للعنونة يُحدد بالعوامل المترابطة التالية :

- عدد أساليب العنونة : بعض أساليب العنونة يمكن التعبير عنها ضمنيا ، فمثلا بعض رموز التعليمات قد تستخدم دائما أسلوب الفهرسة ، وفي بعض الحالات الأخرى أساليب العنونة يجب أن تكون صريحة لذلك يجب أن تخصص مجموعة من الخانات لهذه الأساليب.
- عدد المعاملات : كما ناقشنا سابقا ، فعناوين أقل يمكن أن تجعل البرنامج أطول وغير ملائم . أن التعليمات النموذجية فى المعالجات المعاصرة توفر معاملين . وعنوان كل معامل فى التعليمة ربما يتطلب دليله الخاص (يحدد أسلوب عنونته) أو أن إستخدام دليل العنونة (مُحدد أسلوب العنونة) يكون مقتصراً فقط على أحد حقول العنونة.
- مسجلات مقابل ذاكرة : المعالج يجب أن تكون به مسجلات بحيث يتم أحضار البيانات لداخل المعالج للمعالجة . وبوجود مسجل واحد عام (المجمع) معامل واحد موجود ضمنيا و لا حاجة لأى خانة فى التعليمة للمعامل (للإشارة إليه) . عموما ، البرمجة بوجود مسجل واحد غير ملائمة وتتطلب تعليمات عديدة ، وبوجود عدة مسجلات كانت الحاجة لعدد محدود من الخانات لتحديد المسجل والدراسات أوضحت أن عدداً من 8 – 32 مسجل متعدد الأغراض مناسب لذلك .

- **عدد أطقم المسجلات** : أغلب المعالجات لها طقم واحد من المسجلات **المتعددة الأغراض** حيث يمكن إستخدامها لحفظ البيانات وحفظ العناوين **في الإزاحة** . ولكن بعض المعالجات لها عدة أطقم من المسجلات كل منها مخصص لغرض معين . وميزة الأسلوب الأخير انه بعدد ثابت من المسجلات فإن تجزئة الوظائف بين الأطقم تتطلب إستخدام عدد أقل من الخانات في التعليمة . فمثلاً ، طقمين من المسجلات كل منها مكون من 8 مسجلات يحتاج الى 3 خانات لتعريف كل مسجل ضمن الطاقم ، لذلك رمز العملية أو صيغة المسجل سوف تحدد أي طقم من المسجلات تتم الأشارة عليه .
- مدى العنونة : بالنسبة للعناوين التي تؤشر لمواقع بالذاكرة فمدى العناوين التي يمكن إستخدامها مرتبط بعدد خانات العنوان ، ولأن هذا يفرض قيوداً شديدة فلذلك العنونة المباشرة نادرة الإستخدام . ولكن عند إستخدام العنونة بالإزاحة فمدى العنونة يزداد حسب عرض مسجل العناوين ، ورغم ذلك فإستخدام مقدار كبير من الإزاحة عن العنوان الذى بمسجل العناوين يتطلب عدداً أكثر من الخانات تخصص للعناوين في التعليمة .

5.5 لغة التجميع (الرموز الأبجدية)

يمكن للمعالج فهم وتنفيذ تعليمات الآلة ، وهذه التعليمات هي مجرد الأرقام الثنائية المخزنة في الحاسب ، فإذا رغب المبرمج بالبرمجة مباشرة بلغة الآلة سيكون من الضروري إدخال البرنامج كبيانات ثنائية . مثلاً ، معادلة بسيطة مكتوبة بلغة البيسك N=I+J+K ، ولنفترض أننا نرغب في برمجة هذا المعادلة بلغة الآلة

وتهيئة المتغيرات I ، J ، K بالقيم 2 ، و3 ، و4 على التوالى ، **وهذا موضح في الشكل** (5.8 – أ) . يبدأ البرنامج في الموقع $(101)_{16}$ ، **وتم حجز ذاكرة لمتغيرات** البرنامج الأربعة ابتداء من الموقع $(201)_{16}$. **والبرنامج يتكون من أربعة** تعليمات :

1. تحميل محتويات الموقع 201 الى المسجل المجمع (AC).
2. إضافة محتويات الموقع 202 إلى المسجل المجمع (AC).
3. إضافة محتويات موقع 203 إلى المسجل المجمع (AC).
4. تخزين محتويات المسجل المجمع (AC) في الموقع 204.

من الواضح أن الكتابة بلغة الآلة عرضة للخطأ وعمل ممل جداً . ولذلك إجراء تحسين طفيف في كتابة البرنامج يتم بإستخدام النظام السادس عشرى للإعداد بدلا من النظام الثنائي (الشكل 5.8 – ب) ، ويمكن كتابة البرنامج على شكل سلسلة من السطور كل سطر يحتوي على عنوان موقع بالذاكرة والمناظر السادس عشري للقيمة الثنائية المراد تخزينها في هذا الموقع ، ومن ثم نحن في حاجة إلى برنامج يقبل هذه المدخلات (المناظر السادس عشري) ، ويترجم كل سطر إلى ما يناظره بالثنائي ويخزنه في الموقع المحدد . لمزيد من التحسين يمكننا الإستفادة من الإسم الرمزي لكل تعليمة بحيث يُمكننا ذلك من كتابة برنامج رمزي كما هو مبين في الشكل (الشكل 5.8 – ج) . ففى البرنامج الرمزى كل سطر يتكون من ثلاثة حقول تفصل بينها مسافات ، والحقل الأول يحتوي على عنوان الموقع (الذى تحفظ به التعليمة فى الذاكرة) والحقل الثاني يحتوي على رمز من ثلاثة أحرف لرمز العملية وإذا كانت التعليمة تؤشر للذاكرة الحقل الثالث يحتوي على العنوان . ولتخزين البيانات في مكان ما بالذاكرة فقد ابتكارنا رمزاً زائفاً (DAT)

وهو مجرد إشارة إلى أن الحقل الثالث فى السطر يحتوي على عدد سادس عشري ليتم تخزينه في الموقع المحدد في الحقل الأول . لهذا النوع من المدخلات نحن بحاجة إلى برنامج معقد قليلا ، وهذا البرنامج يقبل سطراً من المدخلات وينتج عدد ثنائي على أساس الحقل الثاني والثالث (إذا كان موجوداً) ، ويخزنها في الموقع المحدد من قبل الحقل الأول . إن إستخدام برنامج رمزي يسهل الأمر ولكن مازال فى الأمر صعوبة ، فإذ يجب أن نعطي عنوانا مطلقاً لكل كلمة (نبدأ تعداد العناوين من عنوان محدد) ، وهذا يعني أنه يمكن تحميل البرنامج والبيانات فى مكان واحد فقط في الذاكرة ، ويجب علينا أن نعرف ذلك المكان مسبقا . والأسوأ من ذلك لنفترض أننا نود أن نغير البرنامج في يوم من الأيام عن طريق إضافة أو حذف السطر فسيؤدي ذلك إلى تغيير عناوين جميع الكلمات التالية .

هناك نظام أفضل بكثير، وهو إستخدام عناوين رمزية ، ويتضح هذا في الشكل (5.8 – د) حيث كل سطر ما زال يتألف من ثلاثة حقول ، الحقل الأول لا يزال للعنوان ، ولكن يتم إستخدام رمز بدلاً من العنوان العددي المطلق . وبعض السطور ليس لها عنوان مما يعني ضمناً أن عنوان هذا السطر هو العنوان التالى للسطر السابق ، وللتعليمات التى تؤشر للذاكرة الحقل الثالث يحتوي أيضا على عنوان رمزي .

مع التعديل الأخير أصبح لدينا لغة التجميع ، ويتم ترجمة البرامج المكتوبة بلغة التجميع (برامج التجميع) إلى لغة الآلة من قبل المترجم التجميعى . وهذا البرنامج ليس فقط للترجمة الرمزية كما ناقشنا سابقا ولكن أيضا تحويل بعض عناوين الذاكرة إلى عناوين رمزية مناظرة.

العنوان	المحتويات			
101	0001	0000	0010	0010
102	0010	0000	0010	0001
103	0011	0000	0010	0011
104	0100	0000	0010	0011
201	0010	0000	0000	0000
202	0011	0000	0000	0000
203	0100	0000	0000	0000
204	0000	0000	0000	0000

(أ) – برنامج بالنظام الثنائى

العنوان	المحتويات
101	2201
102	1202
103	1203
104	3204
201	0002
202	0003
203	0004
204	0000

(ب) – برنامج بالنظام السادس عشري

العنوان	التعليمة
101	LDA 201
102	ADD 202
103	ADD 203
104	STA 204
201	DAT 2
202	DAT 3
203	DAT 4
204	DAT 0

(ج) – برنامج رمزي

العلامة	المعامل	العملية
FORMUL	I	LDA
	J	ADD
	K	ADD
	N	STA
I	2	DAT
J	3	DAT
K	4	DAT
N	0	DAT

(د) – برنامج تجميعي

الشكل (5.8) – حساب المعادلة N=I+J+K

مصطلحات مهمة

العنوان الحقيقي	Real address
العنوان الظاهري	Virtual address
فوري	Immediately
مؤشر صريح	Explicit reference
ظاهري	Virtual
ضمني	Implicit
تنسيق/تشكيل	Format
مقارنة	Compare
لغة الآلة	Machine-language
إزاحة	Shift
أدخال/إخراج معزول مبرمج	Isolated programmed I/O
صلاحية	Privilege
الذاكرة المسقطة	Memory mapped
التخطي	Skip
نقل السيطرة	Transfer-of-control
عودة	Return
السادس عشري	Hexadecimal
الإزاحة	Displacement
مقطع	Segment
مدى العناوين	Address range
أساليب العنونة	Addressing modes
تخصيص	Allocation
قاعدة	Base
قاعدة بالإزاحة	Base with Displacement
العنونة الفورية	Direct addressing
مسجل الفهرسة	Index register
العنوان الفعلي	Effective address
نسبي	Relative
عنونة بالمسجل غير المباشرة	Register Indirect Addressing
العنونة بالازاحة	Displacement Addressing
الفهرسة	Indexing
العنونة بالمسجل	Register addressing
خطي	Linear
تمثيل رمزي	Symbolic representation

تعليمة الآلة	Machine instruction
طقم تعليمات	Instructions set
معامل المصدر	Operand Source
معامل الناتج	Result operand
جلب	Fetch
استدعاء	Call
موقع	Location
مؤشر	Reference
مجمع	Accumulator
تدوير	Rotate
الوصول المباشر للذاكرة	Direct Memory Access (DMA)
نظام التشغيل	Operating system
إستدعاء الإجراء	Procedure call
تفرع	Branch
مشروط	Conditional
العنونة الفورية	Immediate addressing
الفهرسة	Indexing
العنونة غير المباشرة	Indirect addressing
صيغة أو شكل التعليمة	Instruction format
العنونة المباشرة	Direct addressing
العنونة بالإزاحة	Displacement addressing
العنوان الفعلي	Effective address
العنونة بمسجل-القاعدة	Base-register addressing
العنونة بالمسجل	Register addressing
العنونة بالمسجل غير المباشرة	Register indirect addressing
العنونة النسبية	Relative addressing
كلمة	Word
برنامج رمزي	Symbolic program
لغة التجميع	Assembly language
برنامج تجميعي	Assembly programs
عنوان رمزي	Symbolic addresses
علامة	Lable
وحدة إدارة الذاكرة	Memory Management Unit (MMU)

أسئلة للمراجعة

1.- أشرح دورة التعليمة ؟
2.- ما هي أنواع المعاملات في طقم تعليمات المعالج ؟
3.- لماذا نحتاج لتعليمة لنقل السيطرة ؟
4.- ما هي المواقع أو المناطق التى تحتوى على المعاملات ؟
5.- إذا كانت تعليمة تحتوي على أربعة حقول ، ماهو الغرض من كل حقل؟
6.- إشرح بإيجاز القضايا المهمة التى تراعى فى تصميم طقم تعليمات المعالج.
7.- ما هي أنواع المعاملات في طقم تعليمات المعالج ؟ (مثال)
8.- ما هي العناصر النموذجية لتعليمة المعالج وكم العدد الممكن للمعاملات بها ؟
9.- ما هو تأثير عدد المؤشرات للمعامل (0 ، 1 ، 2 ، 3) على نموذج تعليمة المعالج ؟ (وضح بمثال)
10.- ماهى المكونات التى يكمن أستنباطها من طول و تنسيق التعليمة؟ (وضح بمثال)
11.- ماهى الطريقتين لإختبار الحالة فى تعليمة التفرع المشروط ؟
12.- ماهى المسائل التى ينبغى مراعاتها فى تصميم طول نسق التعليمة؟
13.- أشرح الفرق بين تعليمة الإزاحة والتدوير؟
14.- أين يتم تخزين عنوان العودة عند إستدعاء الإجراء.
15.- كثير من أطقم التعليمات بها تعليمة (NOP) و تعنى لا عملية وهى لا تؤثر على حالة المعالج إلا بزيادة عداد البرنامج ، إقترح أى إستخدام لهذه التعليمة .
16.- يمكن إعتبار الإزاحة الحسابية والمنطقية لليسار مناظرة للضرب في 2 عند عدم إحتساب الفيض ، وعند الأخذ في الإعتبار الفيض فالإزاحة الحسابية والمنطقية لليسار تنتج نتيجة مختلفة ولكن الإزاحة الحسابية لليسار تحافظ على إشار العدد ، وضح صحة ماسبق علي عدد صحيح بخمس خانات ثنائية بتمثيل المكمل الثاني.

17.- أفترض أن حقل العنوان فى التعليمة يحتوى على القيمة العشرية (14) ، فأين يكون موقع المعامل المناظر فى حالة :
أ.- العنونة الفورية .
ب.- العنونة المباشرة .
ت.- العنونة غير المباشرة .
ث.- العنونة بالمسجل .
ج.- العنونة غير المباشرة بالمسجل .

18.- أفترض أن معالج مصمم بشكل أن تعليماته تعتمد على المكدس فى تنفيذها ، وهى تتضمن عمليتى (PUSH) و(POP) ، العمليات الحسابية التى ينفذها المعالج تتضمن العنصر أو العنصرين الذين فى قمة المكدس ، نفترض أننا بدئنا بمكدس لا يحتوى على أى عنصر (فارغ) ، ماهى العناصر المتبقية فى المكدس بعد تنفيذ التعليمات التالية :

```
PUSH 4
PUSH 7
PUSH 8
ADD
PUSH 10
SUB
MUL
```

19.- أفترض أن العنوان المخزن في عداد البرنامج يرمز له $س_1$. وحقل العنوان فى التعليمة (مؤشر المعامل) التي في الموقع $س_1$ به القيمة $س_2$ ، المعامل المطلوب تنفيذه من قبل التعليمة مخزن في الذاكرة بالموقع $س_3$ ومسجل الفهرس يحتوى على القيمة $س_4$ ، فماهى العلاقة مابين هذه القيم الرمزية إذا أستعملنا أساليب العنونة التالية :
أ- مباشرة
ب- غير المباشرة
ت- الإزاحة بالفهرسة
ث- الإزاحة النسبية

20.- تعليمة تفرع بأسلوب عنونة نسبي طولها 3 ثمان ، عنوان التعليمة $(256028)_{10}$ ، حدد العنوان المستهدف للتفرع اذا كان مقدار الازاحة المنصوص عليه في التعليمة $(-31)_{10}$؟

21.- نظام الحاسب (IBM 370) لا يوجد به أسلوب العنونة غير المباشرة ، وأفترض أن عنوان المعامل موجود في الذاكرة الرئيسية ، فكيف سيتم التواصل مع المعامل ؟

22.- هل يوجد أي تعليل منطقى لتعليمة بها رمزين لعمليتين ؟

23.- المعالج (Zilog - Z8001) مصمم بتقنية 16-خانة وله تنسيق عام للتعليمة كالتالي :

15–13	13–8	8–7	7–3	3–0
الوضع	رمز العملية	ث/ك	المعامل 2	المعامل 1

حقل الوضع : يحدد كيفية الحصول على المعاملات من حقول المعامل .
حقل ث/ك : يستخدم من بعض التعليمات لتحديد طول المعامل ثمان (8 خانات) أو كلمة (16 خانة) .
حقل المعامل 1 : قد يحدد أحد من 16 مسجل للأغراض العامة (بناءً علي محتوى حقل الوضع).
حقل المعامل 2 : قد يحدد أي من مسجلات الإغراض العامة عدا المسجل 0 ، وعندما تكون قيمة هذا الحقل كلها أصفار فإن رمز العملية يأخذ معني جديد .

أ- كم عدد رموز العمليات التي يوفرها المعالج (Z8001) .
ب- أقترح طريقة فعالة لزيادة عدد رموز العمليات مبيناً المفاضلة الداخلة في المقترح .

24.- **أفترض أن ذاكرة نظام تحتوى على القيم التالية وبعنونة فردية ومسجل مجمع (Acc) ومسجل عام R يحتوى على القيمة 2011 ، بتنفيذ التعليمات التالية ماهى القيم التى سيتم تحميلها للمجمع وماهو العنوان الفعلى المستخدم للمعامل:**

ذاكرة النظام: العنوان	ذاكرة النظام: المحتوى	التعليمات
1011	4011	1- LOAD IMMEDIATE 2011
2011	5011	2- LOAD DIRECT 2011
3011	6011	3- LOAD INDIRECT 2011
4011	7011	4- LOAD Register R
5011	8011	5- LOAD DIRECT 1011
		6- LOAD INDIRECT (1011)
6011	9011	7- LOAD Register R INDIRECT
7011	7011	8- LOAD Displacement R 4000

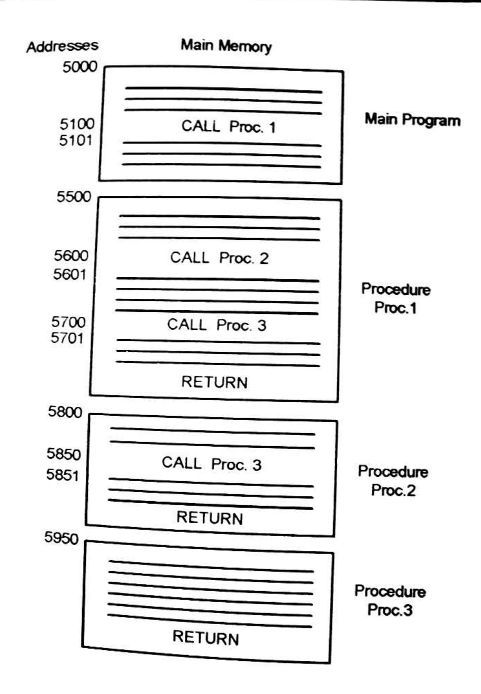
25.- قارن أربع معالجات مختلفة المؤشرات (العناوين) للمعامل (0 ، 1 ، 2 ، 3) بكتابة برنامج لحساب المعادلة :

$$X = (A + B X C)/(D - E X F)$$

لكل معالج والتعليمات المتوفرة للإستخدم لكل معالج هى :

0 - عنوان	1 - عنوان	2 - عنوان	3 - عنوان
PUSH M	LOAD M	MOVE (X ← Y)	MOVE (X ← Y)
POP M	STORE M	ADD (X ← X + Y)	ADD (X ← Z + Y)
ADD	ADD M	SUB (X ← X - Y)	SUB (X ← Z - Y)
SUB	SUB M	MUL (X ← X × Y)	MUL (X ← Z × Y)
MUL	MUL M	DIV (X ← X / Y)	DIV (X ← Z / Y)
DIV	DIV M		

26.- **الشكل التالى يوضح الإجراءات المتداخلة لكيفية إستدعاء الإجراء والعودة منه فى برنامج ما ، أرسم الشكل الذى يبين تسلسل تنفيذ هذا البرنامج موضحا فيه العناوين.**





الفصل السادس

المعالج :-
البنية و الوظيفة



6 – المعالج :- البنية والوظيفة

فى هذا الفصل سوف نتطرق لتنظيم المعالج ، وكذلك سندرس المسجلات التى تشكل الذاكرة الداخلية للمعالج ، وسنعود أيضا لمناقشة دورة التعليمة . وننهى هذا الفصل بدراسة عامة لتقنية التعليمة المجزئة المستخدمة فى المعالجات الحديثة .

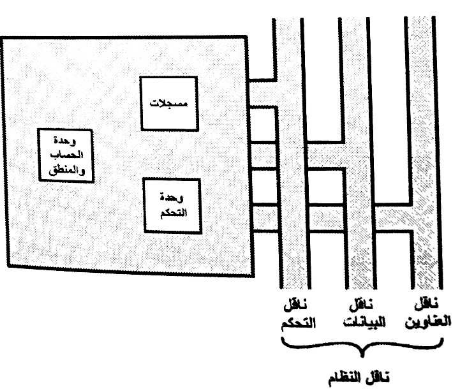
6.1 تنظيم المعالج

لفهم تنظيم المعالج يجب أن ننظر في المتطلبات المفروضة على المعالج بمعنى الأشياء التي يجب علي المعالج أن يفعلها :

1. إحضار التعليمة : المعالج يقرأ التعليمة من الذاكرة (مسجل ، أو الذاكرة الرئيسية) .
2. تفسير التعليمة : يتم فك شفرة التعليمة لتحديد ما يلزم من إجراءات .
3. إحضار البيانات : تنفيذ التعليمة قد يتطلب قراءة بيانات من الذاكرة أو وحدة الإدخال/الإخراج .
4. معالجة البيانات : تنفيذ التعليمة قد يتطلب إجراء بعض العمليات الحسابية أو المنطقية على البيانات.
5. كتابة البيانات : نتائج عملية التنفيذ قد تتطلب كتابة بيانات (نتائج) فى الذاكرة أو وحدة الإدخال/الإخراج .

للقيام بهذه الوجبات ينبغي أن يكون واضحا أن المعالج يحتاج إلى تخزين مؤقت لبعض البيانات ، ويجب أن يتذكر مكان وجود آخر تعليمة سابقة حتى يتمكن من

معرفة مكان الحصول على التعليمة التالية . والمعالج يحتاج لتخزين التعليمات والبيانات بشكل مؤقت أثناء تنفيذ التعليمة وبعبارة أخرى فإن المعالج يحتاج إلى ذاكرة داخلية صغيرة ، والشكل (6.1) يوضح شكلاً مبسطاً للمعالج مشيراً إلى روابطه مع بقية النظام عبر ناقل النظام . ونذكر القارئ بأن مكونات المعالج هى وحدة الحساب والمنطق (ALU) و وحدة التحكم (CU) . فوحدة الحساب والمنطق هى إلتى تقوم فعليا بالحسابات أو معالجة البيانات ، و وحدة التحكم تتحكم فى حركة البيانات والتعليمات من و إلى المعالج كذلك تتحكم فى عمل وحدة الحساب والمنطق ، والشكل يوضح أيضا ذاكرة داخلية خاصة بالمعالج تتكون من فئة من مواقع التخزين تسمى المسجلات .



الشكل (6.1) – وحدة المعالجة المركزية مع ناقل النظام

الشكل (6.2) يوضح نسخة أكثر تفصيلا للمعالج ، والشكل يبين مسارات نقل البيانات والتحكم ، والناقل الداخلي للمعالج الذى يحتاج إليه المعالج لنقل البيانات

بين المسجلات المختلفة بالمعالج و وحدة الحساب والمنطق وذلك **لأن وحدة** الحساب والمنطق تعمل فقط على البيانات التى في **الذاكرة الداخلية للمعالج** (المسجلات) ، ويوضح الشكل أيضا العناصر الأساسية **النموذجية لوحدة** الحساب والمنطق . لاحظ التشابه بين الهيكل الداخلي للحاسب **ككل والبنية** الداخلية للمعالج ، ففي كلتا الحالتين **هناك مجموعة صغيرة من العناصر الرئيسية** التى تكون النظام (الحاسب : معالج و وحدة أدخال/إخراج و ذاكرة ، **المعالج** : وحدة التحكم و وحدة الحساب والمنطق ومسجلات) **متصلة ببعضها بواسطة** مسارات للبيانات .

الشكل (6.2) – البنية الداخلية لوحدة المعالجة المركزية

6.2 تنظيم المسجلات

داخل المعالج هناك مجموعة من المسجلات التي تعمل فى مستوى من الذاكرة أعلى من الذاكرة الرئيسية والذاكرة السريعة في التسلسل الهرمي للذاكرة .
المسجلات في المعالج تؤدى دورين :

1. مسجلات عامة : تمكن المبرمج بلغة الآلة أو التجميع من تقليل الأشارة والتعامل مع الذاكرة الرئيسية وذلك عن طريق الإستخدام الأمثل لهذه المسجلات فى إنجاز المهام والأعمال وهى مرئية للمستخدم بحيث يمكنه التعامل معها بحرية فى تغيير محتوياتها وإستعمالها فى عمليات المعالجة.
2. مسجلات التحكم والمراقبة : تستعمل من قبل وحدة التحكم للسيطرة على عمل المعالج ومراقبة المعالجة وكذلك من قبل بعض برامج نظام التشغيل للتحكم في تنفيذ البرامج .

6.2.1 المسجلات العامة

هذه المسجلات هى التى يمكن الاشارة إليها عن طريق التعليمات التى ينفذها المعالج وهى مرئية للمستخدم ، ويمكننا تصنيفها الى الفئات التالية :

- **مسجلات الأغراض العامة** : يمكن إستخدامها من قبل المبرمج لمجموعة متنوعة من الوظائف ، وأحيانا إستخدامها ضمن طقم التعليمات مناظراً لعملية ما ، بمعنى أن المسجل العام يمكن أن يحتوي على معامل لتعليمة ما . ومع ذلك قد توجد بعض القيود ، فمثلا توجد بعض المسجلات مخصصة لعمليات النقطة العائمة وعمليات المكدس .

- **مسجلات البيانات** : ويمكن إستخدامها فقط لحفظ البيانات و لا يمكن أن تستخدم في حساب عنوان المعامل .
- **مسجلات العنوان** : يمكن أن تكون متعددة الأغراض أو قد تخصص لنوع أو صنف خاص من أساليب العنونة ، ومن الأمثلة على ذلك مسجلات **الفهرسة** وتستخدم للعنونة بالفهرسة ، ومسجل **المكدس** وتستخدم إذا تم توفير أسلوب عنونة بالمكدس للمستخدم حيث يشير المسجل دائما الى قمة المكدس مما يلغى الحاجة للإشارة لقمة المكدس فى التعليمة أثناء إجراء أى عملية على المكدس (دفع ، سحب أو غيره).
- **رموز الحالة** (الأعلام) : مسجلات رموز الحالة وهي جزئيا متاحة للمستخدم ، وبها خانات يضع قيمتها (1/0) المعالج نتيجةً لعملية معالجة. على سبيل المثال ، قد يكون ناتج عملية حسابية ما عدد موجب أو سالب أو صفر أو فيض ، فبلإضافة إلى تخزين قيمة الناتج في مسجل أو الذاكرة، يتم أيضا وضع رمز لحالة الناتج ، فربما في وقت لاحق يتم أختبار رمز الحالة كجزء من عملية تفرع مشروط حيث يتم فحص هذا العلم (رمز لحالة) لإتخاذ قرار التفرع لعنوان محدد . وهذه الرموز قد تجمع و توضع فى مسجل أو مجموعة مسجلات وعادة ما توضع فى مسجل المراقبة . وعموماً ، تعليمات المعالج تسمح بقراءة خانات هذه الرموز فقط (لا تسمح بتغيير قيمتها) عبر الإشارة إليها ضمنيا فى التعليمة .

فى بعض المعالجات فإن إستدعاء روتين جزئى ينتج عنه حفظ تلقائي لجميع محتويات المسجلات العامة (المرئية للمستخدم) من ثم إسترجاعها عند العودة من

تنفيذ هذا الروتين . وفى حين إنه فى بعض المعالجات الاخرى هى مسئولية المبرمج فى حفظ محتويات المسجلات العامة المستخدمة فى البرنامج قبل إستدعاء وتنفيذ الروتين الجزئى وذلك بوضع تعليمات لذلك ضمن البرنامج الجزئى (أول تعليمات فى البرنامج الجزئي - الروتين - مكون من مجموعة تعليمات لحفظ بيانات المعالج الحالية قبل الشروع فى تنفيذ تعليمات الروتين) .

توجد بعض القضايا التصميمية إلتى تخص المسجلات العامة ويجب الإشارة إليها فى هذا الجزء . أولا ، هل نجعل إستخدام المسجلات متعدد الأغراض (عام) أو نحدد ونخصص إستخدامها (خاص) ، وحيث أن ذلك يؤثر على مجمل تصميم طقم تعليمات المعالج . فابستخدام مسجلات خاصة يمكن الإشارة الى مسجل المعامل (المحتوى علي قيمته) ضمنيا فى رمز العملية مما يوفر فى عدد خانات التعليمة ولكن هذا الأسلوب يحد من مرونة البرمجة . ثانيا ، عدد المسجلات سواء كانت عامة أو خاصة بالبيانات أو العناوين حيث أن ذلك يؤثر فى تصميم طقم التعليمات وذلك لأن عدداً أكثر من المسجلات يتطلب عددا أكثر من الخانات للإشارة اليها ، وعدد قليل من المسجلات ينتج عنه كثرة الأشارة أو التعامل مع الذاكرة ، وقد دلت الدراسات أن الأفضل أن يكون عدد المسجلات مابين 8 – 32 مسجل . وفى النهاية ، قضية عرض المسجل (عدد خاناته) ، فالمسجل الذى يتعامل مع العناوين يجب أن يكون بعرض يسع العنوان بالكامل (يسع أكبر عنوان) ، والمسجل الذى يتعامل مع البيانات يجب أن يتمكن من حفظ البيانات باغلب انواعها ، وأخيراً ، بعض المعالجات توفر إمكانية إستعمال مسجلين كمسجل واحد ولكن بعرض مزدوج (بإستخدام تعليمات خاصة بذلك) .

6.2.2 مسجلات التحكم والمراقبة

هناك مجموعة متنوعة من مسجلات المعالج إلتي تستخدم للسيطرة على عمل المعالج ومراقبته ، ومعظمها غير مرئية للمستخدم ، والبعض منها قد تكون واضحة لكن لتعليمات معينة تنفذ فى وضع خاص . طبعا ، كل معالج له تنظيم وتسمية خاصة لمسجلاته ، وسوف نسرد هنا أنواع هذه المسجلات مع وصف موجز لها متخذين نظام الحاسب (IAS) كنموذج ، و لهذا النموذج أربع مسجلات أساسية لتنفيذ إى تعليمة :

- عداد البرنامج (PC) : يحتوي على عنوان موقع التعليمة المطلوب جلبها (إحضارها) للمعالج .
- مسجل التعليمة (IR) : يحتوي على التعليمة التى تم إحضارها حاليا .
- مسجل عنوان الذاكرة (MAR) : يحتوي على عنوان موقع في الذاكرة (يؤشر لموقع) .
- مسجل الذاكرة المؤقت (MBR) : يحتوي على كلمة واحدة من البيانات المراد كتابتها فى الذاكرة أو تمت قراءتها من الذاكرة .

ليس لكل المعالجات مسجلات داخلية مخصصة كمسجل عنوان الذاكرة (MAR) ومسجل الذاكرة المؤقت (MBR) ، ولكن بالضرورة لها آلية تخزين مؤقت للخانات المراد نقلها عبر ناقل النظام أو إستلامها من ناقل النظام .

نموذجيا ، يقوم المعالج بتحديث عداد البرنامج (PC) بعد جلب التعليمة مباشرة بحيث أن عداد البرنامج يشير إلى التعليمة التالية المراد تنفيذها ، ونشير هنا أن تعليمة تفرع أو قفز قد تغير محتوى عداد البرنامج . والتعليمة التى تم جلبها تُحمل فى مسجل التعليمة حيث يتم تفسير رمز العملية و مؤشر المعامل . ويتم تبادل

البيانات مع الذاكرة بإستخدام مسجل عنوان الذاكرة ومسجل الذاكرة المؤقت . ومسجل عنوان الذاكرة مرتبط مباشرة بناقل العناوين ومسجل الذاكرة المؤقت مرتبط مباشرة مع ناقل البيانات ، ويمكن أيضا إستخدام المسجلات العامة لتبادل البيانات مع مسجل الذاكرة المؤقت .

المسجلات الأربع السالفة الذكر تستخدم لحركة البيانات مابين المعالج والذاكرة . وداخل المعالج يجب تقديم البيانات إلى وحدة الحساب والمنطق (ALU) للمعالجة، ولذلك وحدة الحساب والمنطق يمكن أن تتواصل مباشرة مع مسجل الذاكرة المؤقت والمسجلات العامة ، ويمكن أيضاً أن تكون هناك مسجلات عامة لها إمكانية التواصل مع وحدة الحساب والمنطق و كذلك التواصل مع مسجلات أخرى .

العديد من تصاميم المعالجات يتضمن مسجل أو مجموعة من المسجلات للمراقبة. وهذه المسجلات تحتوى على الأعلام التى تشير الى حالة الناتج من آخر عملية معالجة تمت (ليس قيمة الناتج) ، وهذه الأعلام عبارة عن قيم خانات ترفع أو تخفض (1 أو 0) نتيجة لحالة الناتج ، وكل علم يشير لحالة فهناك علم إذا كانت حالة الناتج موجبة وآخر للحالة السالبة وللحالة الصفر أو فيض وغيره . وخانات الأعلام تحفظ فى مسجل أو مسجلات وتعامل هذه المسجلات كتجمع لمجموعة خانات منفصلة وتعالج منطقيا (كل خانة تعامل منفصلة لذلك يمكن فحص كل علم لوحده بحيث يُمكن ذلك من إتخاذ قرار بناءً علي قيمته) .

المسجلات التى تحفظ الأعلام كثيراً ما تعرف بإسم كلمة حالة البرنامج (PSW) بحيث تحتوي على معلومات حول حالة المعالج ، وعادة ما تحتوي على رموز الحالة بالإضافة إلى غيرها من المعلومات ، وحقول كلمة حالة البرنامج هى :

الإشارة : خانة تحتوي على إشارة قيمة ناتج العملية الحسابية السابقة.

صفر : ترفع عندما يكون ناتج العملية الحسابية السابقة صفراً.

الحمل : ترفع إذا كان ناتج العملية الحسابية المنفذة أسفر عن ترحيل (إضافة) أو إلى إقتراض (الطرح) .

التساوى : ترفع إذا كان ناتج المقارنة المنطقية هي مساواة.

فيض : تستخدم للإشارة إلى فيض حسابي.

تمكين/تعطيل المقاطعة : يستخدم لتمكين أو تعطيل المقاطعات .

المشرف : تشير إلى ما إذا كان المعالج ينفذ التعليمات في وضع المشرف أو وضع المستخدم ، فبعض التعليمات ذات الصلاحيات الخاصة تنفذ في وضع المشرف ، وكذلك توجد مناطق معينة فى الذاكرة يمكن الوصول إليها في وضع المشرف فقط .

6.2.3 مثال : تنظيم مسجلات معالج دقيق

المعالج **موتورولا (MC 68000)** له مسجلات بيانات (D_0-D_7) ومسجلات عناوين (A_0-A_7) ذات 32- خانة كما هو موضح الشكل (6.3 – أ) ، وتنقسم إلى ثمانية مسجلات بيانات وتسعة مسجلات عناوين . الثمانية مسجلات الخاصة بالبيانات تستخدم أساسا للتعامل مع البيانات وكذلك تستخدم فى العنونة كمسجل فهرسة . وعرض هذه المسجلات يسمح لعمليات بعرض 8 و 16 و 32 خانة وذلك حسب ما يحدده رمز العملية . ومسجلات العناوين بعرض 32- خانة ، وأثنان من هذه المسجلات تستخدم أيضا كمؤشر للمكدس ، فإحدها للمستخدمين والأخر لنظام التشغيل وذلك حسب وضع العمل/التنفيذ الحالي . وأثنان من المسجلات له نفس الرقم (7) إلا أنه يمكن إستخدام إحدها فقط فى نفس الوقت.

المعالج (MC 68000) يتضمن أيضا مسجل عداد برنامج بطول 32- خانة و **16- خانة مسجل** الحالات (الأعلام) (Status Register).
فريق عمل موتورولا أراد طقم تعليمات عادى بدون مسجلات للإغراض **الخاصة،** وهذا الأهتمام بكفأة التعليمات من أجل برمجة أفضل قادهم لتقسيم **المسجلات** الى مجموعتين وظيفيتين ، وذلك وفر فى طول الحقل المخصص للإشارة للمسجلات بالتعليمة .

أنتل 8086 أتخذت نهجاً مختلفا لتنظيم المسجلات وكما هو موضح فى الشكل (6.3 – ب) فكل مسجل مخصص لغرض خاص ، ولكن بعض المسجلات قابلة للإستخدام للأغراض عامة . أنتل 8086 لها أربعة مسجلات بيانات بعرض 16 – خانة قابلة للعنونة كمسجل 8 – خانات أو 16 – خانة (المسجلات العامة :- AX,BX,CX & DX) ، وأربعة مسجلات 16 – خانة كمسجلات تأشير وفهرسة (مؤشرات وفهرسة :- SP,BP,SI & DI) . وبعض التعليمات يمكنها إستخدام مسجلات البيانات للأغراض العامة بمعنى أن المسجلات يشار إليها ضمنيا ، فعلى سبيل المثال تعليمة الضرب تستخدم دائما مسجل المجمع . والأربع مسجلات الخاصة بالتأشير تستخدم أيضا ضمنيا في عدد من العمليات ؛ وكل واحد منها يحتوي على بداية مقطع (المقطع عبارة عن جزء من الذاكرة الرئيسية القابلة للعنونة) .
توجد أيضا أربعة مسجلات 16 – خانة خاصة بالمقطع (المسجلات :- CS,DS,SS & ES) ، ثلاث منها مخصصة (ضمنيا) للعمل كمؤشر للمقطع **الذى يحتوى على التعليمة الحالية** (مفيدة لتعليمات التفرع) ، ويوجد مقطع **مخصص للبيانات ، ومقطع مخصص** للمكدس . وهذا التخصيص والإستخدام

الضمني وضع من أجل تصغير ودمج البرنامج لكن على حساب التقليل من المرونة . المعالج أنتل 8086 يتضمن أيضاً مسجل مؤشر التعليمة ومسجل أعلام الحالة والتحكم (Flags & Instruction Pointer) .

وجه المقارنة هنا ، إنه لا يوجد مبدأ مقبول عالميا بشأن أفضل طريقة لتنظيم مسجلات المعالج ، وكما مع التصميم العام لطقم التعليمات والعديد من القضايا الأخرى فى تصميم المعالج لا تزال القضية مسألة مفاضلة و إختيار وحكم .

مسجلات بيانات: D0 D1 D2 D3 D4 D5 D6 D7
مسجلات عناوين: A0 A1 A2 A3 A4 A5 A6 A7 A7´
حالة البرنامج: Program counter, Status register
MC68000 (أ)

مسجلات عامة: AX Accumulator, BX Base, CX Count, DX Data
مؤشرات وفهرسة: SP Stack ptr, BP Base ptr, SI Source index, DI Dest index
مقطع: CS Code, DS Data, SS Stack, ES Extrat
حالة البرنامج: Flags, Instr ptr
8086 (ب)

الشكل (6.3) – مثال على تنظيم مسجلات المعالج

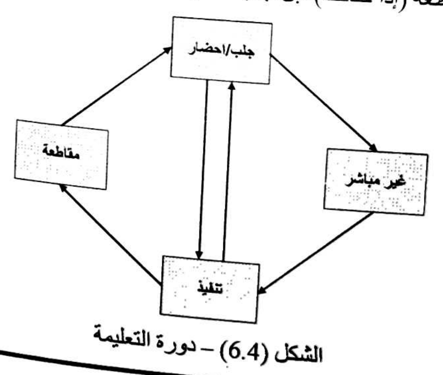
6.3 دورة التعليمة

للتذكير دورة التعليمة تتضمن المراحل الآتية :

- **إحضار/جلب** : قراءة التعليمة التالية من الذاكرة إلى المعالج .
- **تنفيذ** : تفسير شفرة التشغيل (رمز العملية) وتنفيذ العملية المشار إليها .
- **مقاطعة** : إذا تم تمكين المقاطعات وحدثت مقاطعة ،فتحفظ حالة العملية الراهنة ويتم إجراء روتين خدمة المقاطعة .

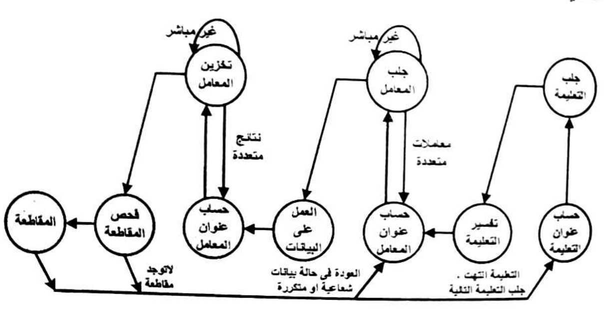
6.3.1 الدورة غير المباشرة

يمكن أن نفكر في جلب العناوين غير المباشرة على إنها مرحلة من مراحل التعليمة كما فى الشكل (6.4) . وكما يوضح الشكل ، فالنشاط الرئيسي يتكون فى التناوب بين نشاط جلب التعليمة و نشاط تنفيذها . فبعد جلب التعليمة يتم فحصها لتحديد ما إذا كان الأمر ينطوي على عنونة غير مباشرة ، فإذا كان الأمر كذلك ، فيتم جلب المعاملات المطلوبة بإستخدام العنونة غير المباشرة ، وبعد التنفيذ تتم معالجة المقاطعة (إذا حدثت) قبل جلب التعليمة التالية .



الشكل (6.4) – دورة التعليمة

الشكل (6.5) يبين هذه الدورة بصورة تفصيلية كاملة وهو يوضح بشكل صحيح طبيعة دورة التعليمة (الحالات التى تمر بها) . فبمجرد ما أن يتم جلب التعليمة ، فلابد من تحديد معاملاتها ومن ثم يتم جلب كل المعاملات من الذاكرة ، وهذه العملية قد تتطلب عنونة غير مباشرة . والمعاملات الموجودة بالمسجلات لا تحتاج لجلب . وبعد تنفيذ شفرة التشغيل (رمز العملية) قد تكون هناك حاجة لعملية مماثلة لتخزين النتيجة في الذاكرة الرئيسية .

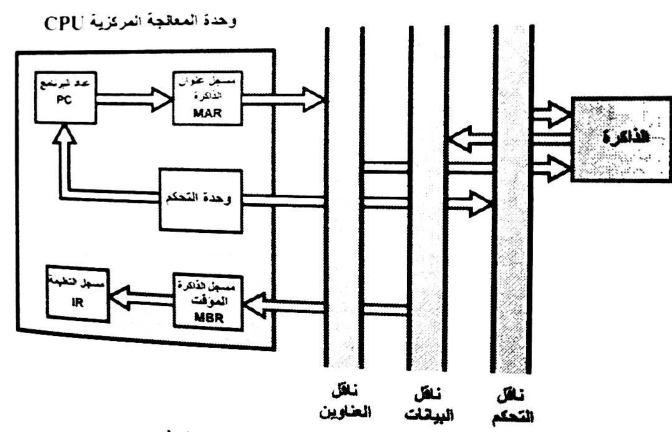


الشكل (6.5) – مخطط للدورة الكاملة لحالات التعليمة

6.3.2 تدفق البيانات

التسلسل الدقيق للأحداث خلال دورة التعليمة يعتمد على تصميم المعالج ومع ذلك يمكننا الإشارة بصورة عامة لما يجب أن يحدث ، ولنفترض هنا أن المعالج يستخدم مسجل عنوان الذاكرة (MAR) ومسجل الذاكرة المؤقت (MBR) وعداد برنامج (PC) ومسجل التعليمة (IR) . فخلال دورة الجلب تتم قراءة التعليمة من الذاكرة ، والشكل (6.6) يوضح تدفق البيانات خلال هذه الدورة . عداد البرنامج يحتوي على عنوان التعليمة التالية التي يتعين جلبها ، ويتم نقل

هذا العنوان إلى مسجل عنوان الذاكرة ويوضع على ناقل العناوين ، ومن ثم **تطلب وحدة التحكم** قراءة الذاكرة ، ويتم وضع النتيجة على ناقل البيانات من ثم **تنسخ فى مسجل الذاكرة المؤقت** ثم تنتقل لمسجل التعليمة . وفي غضون ذلك تتم **زيادة عداد البرنامج** بمقدار = 1 وذلك للتحضير لجلب التعليمة التالية . ومع **إنتهاء دورة الجلب** وحدة التحكم تفحص محتويات مسجل التعليمة لتحديد ما إذا **كان يحتوي على معامل** يستخدم أسلوب العنونة غير المباشرة ، فإذا كان الأمر **كذلك يتم** تنفيذ الدورة غير المباشرة.

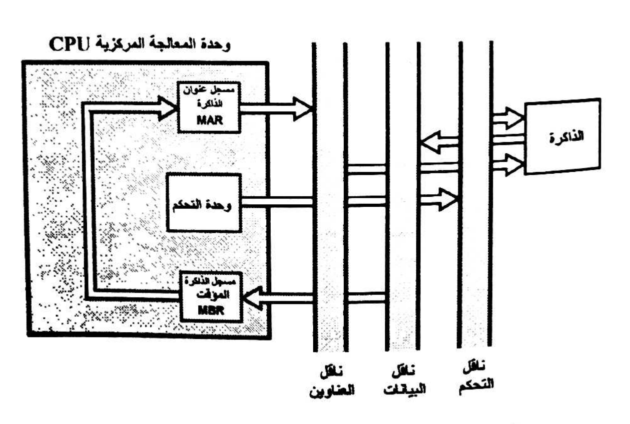


الشكل (6.6) – تدفق البيانات ، دورة الجلب

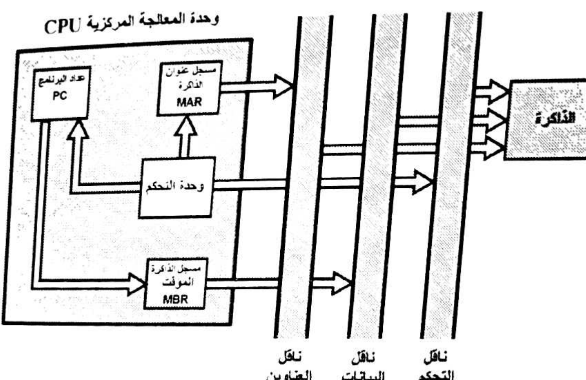
وكما هو مبين في الشكل (6.7) ، سيتم نقل الخانات إلتى فى أقصى اليمين فى مسجل الذاكرة المؤقت وإلتي تحتوي على مؤشر العنوان إلى مسجل عنوان الذاكرة من ثم تطلب وحدة التحكم قراءة الذاكرة وذلك للحصول على العنوان المطلوب للمعامل بحيث يوضع فى مسجل الذاكرة المؤقت (عنونة غير مباشرة).

دورة التنفيذ قد تأخذ أشكالا كثيرة وتعتمد على **التعليمة الموجودة فى مسجل** التعليمة . وهذه الدورة قد تنطوي على نقل البيانات **بين المسجلات ، أو القراءة** أو الكتابة من الذاكرة أو أدخال/إخراج أو **التعامل مع وحدة الحساب والمنطق .**

الشكل (6.8) يوضح دورة المقاطعة حيث يجب أن **يتم حفظ المحتويات الحالية** لعداد البرنامج حتى يتسنى للمعالج إستئناف نشاطه العادي **بعد تنفيذ إجراءات** المقاطعة . وبذلك يتم نقل محتويات عداد البرنامج إلى **مسجل الذاكرة المؤقت** حتى تتم كتابته فى الذاكرة (المكدس) .



الشكل (6.7) – تدفق البيانات ، الدورة الغير مباشرة



الشكل (6.8) – تدفق البيانات ، دورة المقاطعة

6.4 التعليمة المجزئة

مع تطور التقنية أصبح بالإمكان الحصول على أداء عالى للحاسبات ، وبالإضافة الى ذلك إن التحسين فى تنظيم المعالج ساهم ايضا فى الرفع من الأداء . هناك عدة طرق أستعملت فى تنظيم المعالجات منها أستخدام عدة مسجلات بدل **الإقتصار على واحد (المجمع) أو إستخدام الذاكرة السريعة . ومن أحد الأساليب المستخدمة فى تنظيم المعالجات والشائع حاليا هى تقنية التعليمة المجزئة .**

6.4.1 أستراتيجية المعالجة التواردية

فخط التعليمة المجزئة مشابهة لإستخدام خط التجميع في وحدات التصنيع . فخط التجميع يستفيد من حقيقة أن المنتج يمر عبر مراحل مختلفة من الإنتاج ، ففى عملية الإنتاج بإستخدام خط التجميع يمكن العمل على منتج واحد في مراحل

مختلفة في وقت واحد (كل مرحلة تختص بجزء من المنتج) ويطلق على هذا الأسلوب فى الانتاج بخط التوارد ، حيث تستقبل مدخلات جديدة فى بداية خط التجميع قبل أن تظهر المدخلات القديمة فى نهاية خط التجميع . وكمقاربة بسيطة يمكن تقسيم معالجة التعليمة إلى مرحلتين : جلب تعليمة وتنفيذ التعليمة ، ففى أثناء تنفيذ التعليمة لا يوجد تواصل مع الذاكرة الرئيسية ويمكن إستخدام هذا الوقت لجلب التعليمة التالية بالتوازي مع التنفيذ الحالي .

تعليمة ← جلب/احضار ← تعليمة ← تنفيذ ← النتيجة

(أ) - منظر مبسط

انتظار – عنوان جديد – انتظار
تعليمة ← جلب/احضار ← تعليمة ← تنفيذ ← النتيجة
تجاهل

(ب) - منظر مفصل

الشكل (6.9) – التعليمة المجزئة بمرحلتين : منظر مبسط ، منظر تفصيلي

الشكل (6.9 – أ) يصور خط توارد بمرحلتين مستقلتين . **فالمرحلة الأولى لجلب التعليمة وتخزينها مؤقتا والمرحلة الثانية للتنفيذ . وعندما تكون المرحلة الثانية شاغرة تقوم المرحلة الأولى بتمرير التعليمة المخزنة لديها لها . وأثناء تنفيذ المرحلة الثانية للتعليمة تستفيد المرحلة الأولى من الوقت (ليس هناك تواصل مع الذاكرة الرئيسية) لجلب التعليمة التالية وتخزينها مؤقتا وهذا ما يسمى الجلب المسبق للتعليمة أو الجلب المتداخل . لاحظ أن هذا النهج ، والذي ينطوي على**

التخزين المؤقت للتعليمة يتطلب المزيد من المسجلات . وبشكل عام ، خط التوارد يتطلب مسجلات لتخزين البيانات بين المراحل ، وينبغي أن يكون واضحا أن هذه العملية سوف تسرع من تنفيذ التعليمات . وإذا كانت مراحل الجلب والتنفيذ متساوية المدة ، فإنه سيتم إختصار دورة التعليمة للنصف ، ومع ذلك ، فإذا ما نظرنا عن كثب إلى خط التوارد الموضح بالشكل (6.9 – ب) سوف نرى أن هذه المضاعفة فى التنفيذ غير واردة لسببين :

1- وقت التنفيذ عموما أطول من وقت الجلب . فالتنفيذ ينطوي على إجراءات قراءة وتخزين المعاملات وأداء بعض العمليات ، وبالتالي مرحلة الجلب تضطر إلى الإنتظار لبعض الوقت قبل أن تتمكن من تفريغ المخزن فيها.

2- تعليمة التفرع المشروط تجعل عنوان التعليمة التالية التي سيتم جلبها غير معروف ، وبالتالي يجب على مرحلة الجلب الإنتظار حتى تحصل على عنوان التعليمة التالية من مرحلة التنفيذ وبالتالى على مرحلة التنفيذ الإنتظار إلى حين جلب التعليمة التالية .

التخمين يحد من خسارة الوقت للسبب الثاني ، وهناك قاعدة بسيطة هي : عندما يتم تمرير تعليمة تفرع مشروط من مرحلة الجلب إلى مرحلة التنفيذ ، فإن مرحلة الجلب تجلب التعليمة التالية في الذاكرة بعد تعليمة التفرع ، ثم إذا لم ينفذ التفرع لم نضيع الوقت ولكن إذا تم اتخاذ قرار التفرع ، لا بد من تجاهل التعليمة التى تم إحضارها وجلب تعليمة جديدة . ففي حين أن هذه العوامل تقلل من فعالية المحتملة لخط التوارد بمرحلتين ، ولكن بعض التسريع سيتم وللحصول على مزيد من التسريع يجب أن يحتوى خط التوارد على مزيد من المراحل .

التحليل التالي يقدم خط توارد بستة مرحل لمعالجة تعليمة :

- إحضار تعليمة (FI) : إقرأ التعليمة المتوقعة التالية في التخزين المؤقت.
- فك شفرة التعليمة (DI) : تفسير شفرة التشغيل (رمز التعليمة) وتحديد المعاملات.
- حساب المعاملات (CO) : حساب العنوان الفعال لكل معامل مصدر وهذا قد يتضمن حساب أساليب العنونة (مباشر او غير المباشر أو إزاحة أو غيرها من أساليب العنونة) .
- إحضار المعاملات (FO) : جلب كل المعاملات من الذاكرة و المعاملات التى في المسجلات لا حاجة لجلبها.
- تنفيذ التعليمة (EI) : تنفيذ العملية المشار إليها وتخزين النتيجة إن وجدت في الموقع المحدد لذلك .
- كتابة المعامل (WO) : تخزين النتيجة في الذاكرة.

مع هذا التحليل المراحل المختلفة لا تشغل فترات زمنية متساوية تقريبا ، ومن أجل التوضيح لنفترض أنها فترات متساوية . وبإستخدام هذا الأفتراض الشكل (6.10) يوضح أن خط توارد بستة مراحل يمكن أن يقلل من وقت تنفيذ 9 تعليمات من 54 إلى 14 وحدة زمنية من الزمن . والشكل (6.11) يوضح نفس تسلسل المراحل لكن إتجاه الزمن يؤشر إلى الأسفل (رأسيا) وكل صف (أفقيا) يوضح حالة خط التوارد فى وقت ما بحيث يبين تأثير وضريبة تنفيذ تعليمة التفرع على تسلسل عمل خط التوارد . وفى الشكل (6.11 – أ) ، ومقارنة مع الشكل (6.10)، فخط التوارد يمتلئ عند الزمن 6 بواسطة ستة تعليمات مختلفة فى مراحل عمل

مختلفة ، وأستمرت ممتلئة حتى الزمن 9 مع إفتراض أن التعليمة 9 هى آخر تعليمة سوف تنفذ .

وفى الشكل (6.11 – ب) ، فإن خط التوارد يمتلئ عند الزمن 6 و 7 ، ففى الزمن 7 التعليمة 3 فى مرحلة التنفيذ وتنفذ تعليمة تفرع الى التعليمة 15 ، وفى هذه الحالة التعليمات من 4 الى 7 سوف تطرد من خط التوارد ، وعند الزمن 8 تعليمتين فقط فى خط التوارد وهما 3 و 15 . والشكل (6.12) يوضح المخطط الإنسيابى المقترح للمنطق اللازم لخط توارد بستة مراحل يأخذ فى الحسبان التفرع والمقاطعة.

الزمن ←

14	13	12	11	10	9	8	7	6	5	4	3	2	1	
								WO	EI	FO	CO	DI	FI	**تعليمة - 1**
							WO	EI	FO	CO	DI	FI		**تعليمة - 2**
						WO	EI	FO	CO	DI	FI			**تعليمة - 3**
					WO	EI	FO	CO	DI	FI				**تعليمة - 4**
				WO	EI	FO	CO	DI	FI					**تعليمة - 5**
			WO	EI	FO	CO	DI	FI						**تعليمة - 6**
		WO	EI	FO	CO	DI	FI							**تعليمة - 7**
	WO	EI	FO	CO	DI	FI								**تعليمة - 8**
WO	EI	FO	CO	DI	FI									**تعليمة - 9**

EI = تنفيذ التعليمة
FI = احضار التعليمة
FO = احضار المعاملات
DI = تفسير التعليمة
WO = كتابة المعاملات
CO = حساب المعاملات

الشكل (6.10) – البيان الزمنى لعمل التعليمة المجزئة

(أ) - بدون تفرع (الزمن ↓)

WO	EI	FO	CO	DI	FI	
					I_1	1
				I_1	I_2	2
			I_1	I_2	I_3	3
		I_1	I_2	I_3	I_4	4
	I_1	I_2	I_3	I_4	I_5	5
I_1	I_2	I_3	I_4	I_5	I_6	6
I_2	I_3	I_4	I_5	I_6	I_7	7
I_3	I_4	I_5	I_6	I_7	I_8	8
I_4	I_5	I_6	I_7	I_8	I_9	9
I_5	I_6	I_7	I_8	I_9		10
I_6	I_7	I_8	I_9			11
I_7	I_8	I_9				12
I_8	I_9					13
I_9						14

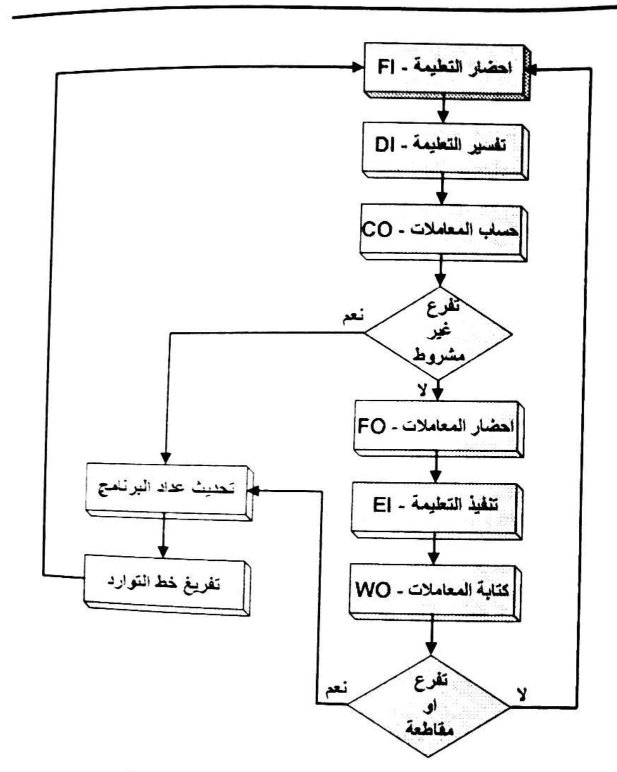
(ب) – تفرع مشروط

WO	EI	FO	CO	DI	FI	
					I_1	1
				I_1	I_2	2
			I_1	I_2	I_3	3
		I_1	I_2	I_3	I_4	4
	I_1	I_2	I_3	I_4	I_5	5
I_1	I_2	I_3	I_4	I_5	I_6	6
I_2	I_3	I_4	I_5	I_6	I_7	7
I_3					I_{15}	8
				I_{15}	I_{16}	9
			I_{15}	I_{16}		10
		I_{15}	I_{16}			11
	I_{15}	I_{16}				12
I_{15}	I_{16}					13
I_{16}						14

الشكل (6.11) – توضيح مغاير لخط التوارد

6.4.2 مخاطر خط التوارد

تحدث أخطار خط التوارد عندما يجب على الخط أو جزء منه **المماطلة/الإنتظار** لأن الظروف لا تسمح بالتنفيذ المستمر (**تسلسل توارد التعليمات**) ، **وهناك ثلاثة** أنواع من المخاطر : **الموارد والبيانات والتحكم .**



الشكل (6.12) – معالجة مجزئة بستة مراحل

1. مخاطر الموارد : تحدث عندما تكون أثنين (أو أكثر) من التعليمات التي هي **بالفعل في خط التوارد في حاجة إلى نفس المورد** (وحدة الحساب والمنطق مثلاً) **والنتيجة هي أنه يجب تنفيذ التعليمات تسلسلياً وليس بالتوازى فى جزء من خط التوارد ، ويشار إليه أحيانا كخطر هيكلي . ومثال على ذلك خط توارد مبسط**

بخمس مراحل (كل المراحل تستغرق زمن متساوى – **نبضة زمنية واحدة**) ، فنموذجياً فى كل نبضة تدخل تعليمة جديدة ، **ومع إفتراض أن الذاكرة لها منفذ** واحد ، وأن جلب التعليمة و قراءة البيانات وكتابتها تُجرى مرة كلّ على جدة ، ومع تجاهل وجود ذاكرة سريعة فإن قراءة أو كتابة بيانات **(للذاكرة الرئيسية) لا** يمكن أن تُجرى بالتوازى مع جلب تعليمة ، والحل فى هذه الحالة أن يكون للذاكرة عدة منافذ يمكن من خلالها التواصل مع محتوياتها .

والشكل (6.13) يوضح مثال على مخاطر الموارد ، **حيث يفترض الشكل أن** معامل المصدر للتعليمة-1 موجود فى الذاكرة **وليس فى مسجل بوحدة المعالجة** المركزية ، ولذلك مرحلة جلب التعليمة في خط التوارد **يجب أن تُعطل لدورة** واحدة قبل البدء بمرحلة جلب التعليمة للتعليمة-3 ، **ويفترض الشكل أن جميع** باقى المعاملات موجودة بالمسجلات .

نبضة الساعة

	1	2	3	4	5	6	7	8	9
تعليمة - 1	FI	DI	FO	EI	WO				
تعليمة - 2		FI	DI	FO	EI	WO			
تعليمة - 3			FI	DI	FO	EI	WO		
تعليمة - 4				FI	DI	FO	EI	WO	

(أ) - خط توارد بخمس مراحل ، الحالة النموذجية

الشكل (6.13 – أ) – **مثال على مخاطر الموارد**

نبضة الساعة ←

	1	2	3	4	5	6	7	8	9
تعليمة - 1	FI	DI	FO	EI	WO				
تعليمة - 2		FI	DI	FO	EI	WO			
تعليمة - 3			معطل	FI	DI	FO	EI	WO	
تعليمة - 4					FI	DI	FO	EI	WO

(ب) – معامل المصدر للتعليمة - I فى الذاكرة

الشكل (6.13 – ب) – مثال على مخاطر الموارد

2. مخاطر البيانات : يحدث عندما يكون هناك صراع في الوصول لموقع معامل ما مابين أثنين (أو أكثر) من التعليمات . وبشكل عام يمكننا أن نقول إن الخطر في هذه الحالة : أثنان من التعليمات في برنامج ما يتم تنفيذها بالتوازى وتطلب الوصول لمعامل ما فى موقع معين من الذاكرة أو فى مسجل ما فى نفس الوقت. ففى حالة أن التنفيذ يتم بتسلسل صارم لن تحدث مشكلة ، ولكن إذا كان التنفيذ يتم فى خط توارد فمن المحتمل أن يتم تحديث قيمة المعامل بحيث تكون النتيجة مغايرة عن التنفيذ فى حالة التسلسل الصارم .

مثال على ذلك ، تسلسل تنفيذ التعليمات المعالج **أنتل x86** التالية :

- ADD EAX, EBX   */ EAX = EAX + EBX
- SUB ECX, EAX   * / ECX = ECX – EAX

فالتعليمة الأولى تجمع محتويات المسجلين 32-خانة (EAX) و (EBX) **وتخزن الناتج في المسجل (EAX) ، والتعليمة الثانية تطرح محتويات المسجل** 32-خانة **(EAX) من (ECX) وتخزن الناتج في المسجل** (ECX) .

التعليمة (ADD) لن تُحدث المحتوى حتى نهاية المرحلة 5 وعند النبضة 5 الساعة . والتعليمة (SUB) تحتاج للقيمة فى بداية المرحلة 2 ، وعند النبضة 4 الساعة . وللحصول على ناتج صحيح خط التوارد يجب أن يُعطل لنبضتى ساعة. وبدون منطق مادى خاص أو خوارزمية معينة لهذه الحالة فإن خط التوارد غير فعال ، والشكل (6.14) يوضح مخاطر البيانات المبينة فى المثال السابق .

نبضة الساعة ←

	1	2	3	4	5	6	7	8	9	10
ADD EAX, EBX	FI	DI	FO	EI	WO					
SUB ECX, EAX		FI	DI	معطل		FO	EI	WO		
تعليمة - 3			FI			DI	FO	EI	WO	
تعليمة - 4						FI	DI	FO	EI	WO

الشكل (6.14) – مثال على مخاطر البيانات

أنواع مخاطر البيانات :

- قراءة بعد كتابة أو الإعتمادية الحقيقية
  - (Read After Write (RAW))
  - تعليمة تُعدل محتوى مسجل أو موقع ذاكرة ، **والتعليمة** التالية تقرأ البيانات التى فى ذلك الموقع ، **فإن الخطر يقع** إذا تمت القراءة قبل الكتابة .
- كتابة بعد قراءة أو الإعتمادية العكسية
  - (Write After Read (WAR))

- تعليمة تقرأ محتوى مسجل أو موقع ذاكرة ، والتعليمة التالية تكتب بيانات في ذلك الموقع ، فإن الخطر يقع إذا تمت الكتابة قبل إنجاز القراءة .

- كتابة بعد كتابة أو إعتمادية الإخراج
  - (Write After Write (WAW))
  - تعليمتان تكتبان لنفس الموقع في الذاكرة ، فإن الخطر يقع إذا تمت الكتابة بعكس التسلسل المفروض .

3. **مخاطر التحكم** : تعرف أيضا باسم مخاطر التفرع ، وتحدث عندما يتخذ خط التوارد قرار تنبؤ بتفرع خاطئ ويجلب بالتالي تعليمة إلى خط التوارد ثم يتم تجاهلها لاحقاً . وتوجد عدة حلول لهذه المعضلة منها :

- خطوط توارد متعددة .
- الجلب المسبق للتفرع المستهدف .
- التنبؤ بالتفرع .
- تأخير التفرع .

مصطلحات مهمة

Instruction Cycle	دورة التعليمة
Instruction Prefetch	الجلب المسبق للتعليمة
Branch Prediction	تنبوء بتفرع
Interrupt Cycle	دورة مقاطعة
Fetch	جلب/إحضار
Indirect	غير المباشر
Fetch Instruction (FI)	إحضار تعليمة
Calculate Operands (CO)	حساب المعاملات
Execute Instruction (EI)	تنفيذ التعليمة
Data Hazard	مخاطر البيانات
Control Hazard	مخاطر تحكم
Instruction Pipeline	التعليمة المجزئة
Pipeline Hazards	مخاطر خط التوارد
Fetch Data	إحضار البيانات
Control Bus	ناقل التحكم
Data Bus	ناقل البيانات
Address Bus	ناقل العنوان
System Bus	ناقل النظام
Decode Instruction (DI)	فك شفرة التعليمة
Fetch Operands (FO)	إحضار المعاملات
Write Operand (WO)	كتابة المعامل
Resource Hazard	مخاطر المورد
Structural Hazard	خطر هيكلي
Pipeline Processing	المعالجة التواردية/المجزئة
Program Status Word (PSW)	كلمة حالة البرنامج

أسئلة للمراجعة

(بالنسبة للمسائل البرمجية الخاصة بالمعالج الأفتراضى (PIPPIN) فيمكن الأطلاع وأستخدام طقم تعليمات المعالج من موقع المعالج على شبكة المعلومات وكذلك يوجد شرح كامل لعمله – روابط المعالج مذكورة في مصادر ومراجع الكتاب ، ويمكن إجراء تطبيق مباشر للبرامج المكتوبة على برنامج المحاكاة الخاص به والموجود بالموقع والتحقق من النتائج).

1. إشرح الدورة الكاملة للتعليمة ؟
2. أفترض أن معالج مصمم بشكل أن تعليماته تعتمد على المكدس فى تنفيذها ، وهى تتضمن عمليتى (PUSH) و(POP) ، والعمليات الحسابية التى ينفذها المعالج تتضمن العنصر أو العنصرين الذين فى قمة المكدس ، ولنفترض أننا بدئنا بمكدس لا يحتوى على أى عنصر (فارغ) ، ماهى العناصر المتبقية فى المكدس بعد تنفيذ التعليمات التالية :

```
PUSH 4
PUSH 7
PUSH 8
ADD
PUSH 10
SUB
MUL
```

3. أذا كانت أخر عملية جرى تنفيذها فى معالج هى (A – B) ، حيث A = 11110000 ، و B = 0010100 ، ماذا ستكون قيم الأعلام التالية : حمل (Carry) ، صفر (Zero) ، فيض (Overflow) ، الأشارة (Sign) .
4. إشرح مخاطر البيانات التى يتعرض لها خط التوارد موضحا انواعه؟
5. أفترض معالج تواردى بأربع مراحل (جلب، تفسير، تنفيذ، تخزين) أرسم الشكل الزمنى الذى يوضح كم وحدة زمنية مطلوبة لتنفيذ (7) تعليمات علما بأن التعليمة رقم (3) قفز الى التعليمة (10) مع عدم وجود بيانات معتمدة على بعضها. (أنظر للشكل 6.11).

6. كيف يكون تدفق البيانات أثناء تنفيذ دورة الجلب/الأحضار و دورة المقاطعة فى المعالج؟
7. ماهى أستراتيجية المعالجة التواردية ، أشرح مستعينا بالمخطط الأنسيابى؟
8. ماهى المسائل التى ينبغى مراعاتها فى تصميم طول نسق التعليمة؟
9. ماهى العوامل التى تحدد أستخدام خانات العنونة ؟
10. ماهى وظيفة لغة التجميع ؟
11. إفترض معالج له تعليمة بطول 16-خانة و تنقسم لجزئين: 6 - خانات الأولى تحتوى على رمز التعليمة فى حين أن بقية الخانات تحتوى على المعامل نفسه (فورى) أو مؤشر له :
    - ماهي أقصى سعة للذاكرة القابلة للعنونة ؟
    - ماهو عدد التعليمات بطقم التعليمات ؟
    - ماهو عرض كل موقع بالذاكرة ؟
    - كم عدد الخانات المطلوبة لكل من مسجل التعليمة و مسجل عداد البرنامج ؟
12. معالج له تعليمة بطول 32-خانة والعناوين بطول 12-خانة . مع إفتراض أن له 250 تعليمة ذات عنوانين (مؤشرين لمعاملين) ، كم عدد التعليمات ذات العنوان الواحد (مؤشر واحد للمعامل) التي يمكن أن تكون بطقم تعليمات هذا المعالج ؟ (وضح أجابتك)
13. معالج يشتغل بمعدل 5 غيغا هرتز نبضة لساعة النظام ، فما طول دورة النبضة و ماهو زمن نوع معين من التعليمات يستغرق 3 دورات نبضية ؟
14. مع أفترض أن تعليمة تستغرق 4 دورات لتنفذ على معالج غير تواردى : دورة لجلب التعليمة ، ودورة لتفسير التعليمة ، ودورة لإجراء العملية الحسابية أو المنطقية ، ودورة لتخزين الناتج . وفي معالج تواردى بأربع مراحل نفس التعليمة تستغرق 4 دورات لتنفذ ، فكيف يمكننا القول أن المعالجة التواردية تُسرع من تنفيذ البرنامج ؟
15. طول التعليمة لتنسيق معين لتعليمة معالج هو 11-خانة و طول حقل العنوان 4-خانات ، هل من الممكن أن يتضمن طقم تعليمات هذا المعالج : 5 تعليمات ذات عنوانين (معاملين) ، و45 تعليمة ذات عنوان واحد و 32 تعليمة ذات

صفر عنوان (بدون معامل) بالتنسق المذكور ؟. ومع أفترض أن مصمم معمارية المعالج صمم 6 تعليمات ذات عنوانين و 24 تعليمة ذات صفر عنوان بإستخدام التنسيق السالف الذكر ، فماهو أقصي عدد من التعليمات ذات العنوان الواحد التي يمكن إضافتها لطقم التعليمات ؟

16. نظام حاسب له وحدة ذاكرة ذات 24-خانة للكلمة . وطقم تعليمات هذا النظام يتضمن 150 عملية مختلفة ، وكل التعليمات لها حقل لرمز العملية و حقل للعنوان . وكل تعليمة تسع كلمة ذاكرة واحدة :

- كم عدد الخانات المطلوبة لرمز العملية ؟
- كم عدد الخانات المتبقية لعنوان المعامل ؟
- كم أقصى سعة للذاكرة يمكن للنظام التواصل معها ؟
- كم أقصى عدد ثنائي صحيح بدون أشارة يمكن ان يوضع في التعليمة ؟
- كم أقصى عدد ثنائي صحيح بدون أشارة يمكن ان يوضع في كلمة بالذاكرة ؟

17. اكتب برامج بإستخدام لغة الرموز للمعالج الأفتراضى (PIPPIN) للتالى :

- **إيجاد ناتج المعادلة :** $X = ((M\times5)+(N-2))/(K\times2)^2$.
- **إيجاد ناتج المعادلة :** $Z = X(X + Y)+Y$.
- **إيجاد ناتج المعادلة :** $X = (5\times W) / (2+Y)$.
- **الى مالا نهاية يضيف قيمة 3 الى W ويخزن النتيجة فى Y .**
- **يحفظ قيمة 5 فى مسجل أذا كانت قيمة X تساوى صفر ويحفظ قيمة 10 أذا كانت قيمة X خلاف ذلك .**
- **عد تنازلى من قيمة 50 الى 0 بخطوة مقدارها 10 .**
- **إيجاد متوسط الاعداد من 0 ، ... ، X .**
- **إيجاد** Y^X .
- **إيجاد النسبة المئوية لقيمة X نسبة الى قيمة Y .**
- **طرح قيمة 3 من W حتى تصل قيمته إلى 0 ثم تخزن الناتج فى Y (W−30). (مستعملاً التفرع المشروط)**

- برنامج يحول درجة الحرارة من نظام المئوى (C) الى النظام الفهرنهايت (F) مستخدماً معادلة التحويل التالية : $C = 5 \times (F - 32) / 9$. علما بأن قيمة F مخزنة فى العنوان W والعنوانين الأخرين للمعالج X ، Y يمكن أستخدامهما فى البرنامج للنتائج الجزئية.

الفصل السابع

نظام الذاكرة والذاكرة السريعة

7- نظام الذاكرة والذاكرة السريعة

المكون الرئيسي الثانى لنظام الحاسب وحسب نموذج فون نيومان لنظام الحاسب هو نظام الذاكرة . وبالرغم من أنها تبدو بسيطة كمفهوم ولكن ذاكرة الحاسب تشمل مدى عريض من الأنواع والتقنيات والتنظيمات والأداء والتكلفة مقارنة بأى مكون آخر فى نظام الحاسب . ولا توجد تقنية واحدة ترضي المطلوب من الذاكرة فى نظام الحاسب ، لذلك فى نظام الحاسب النموذجى توجد مجموعة متسلسلة من الأنظمة الجزئية للذاكرة ، فمنها ماهو داخل الحاسب و أخرى خارجه. وفى هذا الفصل سوف نتناول نظرة عامة لأنظمة الذاكرة بالحاسب ثم أول مستوى فى أنظمة الذاكرة : الذاكرة السريعة .

7.1 نظرة عامة على نظام ذاكرة الحاسب

7.1.1 خصائص نظم الذاكرة

يمكن تسهيل موضوع ذاكرة الحاسب المعقد إذا أمكن لنا أن نصنف أنظمة الذاكرة وفقا لخصائصها الرئيسية وهي كالتالى :

أ. **الموقع** : الموقع يشير إلى ما إذا كانت الذاكرة هي داخل أو خارج الحاسب . وعادة ما تتساوى الذاكرة الداخلية مع الذاكرة الرئيسية ولكن هناك أشكال أخرى من الذاكرة الداخلية فالمعالج يتطلب ذاكرة محلية خاصة على شكل مسجلات ، وذاكرة التخزين السريع هى شكل آخر من أشكال الذاكرة الداخلية . الذاكرة الخارجية تتكون من أجهزة التخزين الطرفية مثل القرص ، والشريط وهي في متناول المعالج عن طريق أجهزة تحكم الإدخال/الإخراج .

2. **السعة** : سعة الذاكرة الداخلية عادة ما يتم التعبير عنها بعدد الخانات (8 خانات ثنائية = ثمان – بايت = 8 بت) أو الكلمات . والطول الشائع للكلمة هو 8 او 16 او 32 خانة ، كذلك يتم أيضا التعبير عن سعة الذاكرة الخارجية بعدد الثمان .

3. **وحدة النقل** : بالنسبة للذاكرة الرئيسية هى عدد الخانات التى تُقرأ من أو تُكتب في الذاكرة في كل مرة (بالتوازى) ، فوحدة النقل لا تساوى بالضرورة كلمة واحدة . وأما بالنسبة للذاكرة الخارجية فغالبا ما يتم نقل البيانات في وحدات أكبر بكثير من كلمة واحدة وتتم الإشارة إليها على أنها قوالب .

4. **طريقة التواصل** : للوصول إلى وحدة معينة من البيانات في الذاكرة هناك عدة أساليب وتشمل التالى :

أ. الوصول المتسلسل : يتم تنظيم الذاكرة فى وحدات من البيانات تدعى سجلات ، والوصل اليها يتم في تسلسل خطي محدد ، فالمعلومات المحفوظة عن موقع السجل تستخدم كعازل بين السجلات . وفى هذه الطريقة تستخدم آلية واحدة (رأس خاص) للقراءة وللكتابة ، ويتم الأنتقال من الموضع الحالى الى المطلوب فى تسلسل خطى وبالمرور على جميع السجلات التى بين السجل الحالى والمطلوب (تتبع) ، ولاحظ هنا أن زمن الوصول لسجل عشوائى متغير بشدة ، وهذه الطريقة تستخدم فى وحدة الأشرطة المغناطيسية .

ب. الوصول المباشر: كما هو الحال مع الوصول المتسلسل ، فالوصول المباشر ينطوي على آلية واحدة للقراءة والكتابة . وفى هذه الطريقة كل قالب أو سجل له عنوان فريد بناءاً على موقعه فى الذاكرة ، ويتم الوصول

أليه مباشرة عن طريق بحث متتابع أو العد أو الأنتظار حتى الوصول إلى الموقع النهائي بدون المرور على السجلات البينية ولكن بأستخدام عنوان السجل وفى هذه الطريقة زمن الوصول لسجل عشوائى متغير ، ويتم إستخدامها فى وحدة الأقراص .

ت. الوصول العشوائي : لكل موقع في الذاكرة عنوان فريد حسب آلية العنونة . وزمن الوصول إلى موقع معين فى الذاكرة لا يعتمد على الوصول لموقع سابق وثابت وبالتالي يمكن أختيار أي موقع عشوائيا والوصول إليه مباشرة ، والبيانات المخزنة في الذاكرة الرئيسية والسريعة الوصول إليها عشوائى .

ث. الترابطى : هذا نوع من ذاكرة الوصول العشوائي التي تُمكن من إجراء مقارنة لخانات الموقع المطلوب مع خانات معينة بغرض التوافق من عدمه وبالتالي يتم إسترداد والوصول للكلمة على أساس جزء من محتوياتها بدلاً من عنوانها ، وبعض أنواع الذاكرة السريعة التواصل مع محتواها يتم بهذه الطريقة .

5. **الأداء** : تستخدم ثلاثة معايير لقياس الأداء :

أ- زمن الوصول (التأخير) : بالنسبة لذاكرة الوصول العشوائي **هو الزمن الذي يُستغرق لإجراء عملية القراءة أو الكتابة ، ويتضمن ذلك الزمن المستغرق من لحظة تقديم العنوان إلى الذاكرة إلى لحظة تخزين البيانات أو إتاحتها للإستخدام . أما بالنسبة لذاكرة الوصول غير العشوائي ، فزمن الوصول هو الزمن الذي يُستغرق لوضع آلية القراءة/الكتابة في الموقع المطلوب .**

ب- زمن دورة الذاكرة : يتكون من زمن الوصول بالإضافة الى أى زمن مطلوب قبل بداية الوصول التالى .

ت- معدل النقل : هو المعدل الممكن لنقل البيانات من أو إلى وحدة الذاكرة . وبالنسبة لذاكرة الوصول العشوائي هو يساوي 1/(زمن دورة الذاكرة).

6. **النوع المادي** : لقد أستخدمت مجموعة متنوعة من الأنواع المادية فى تصميم الذاكرة . والأكثر شيوعا اليوم هي الذاكرة الشبه موصلة (الألكترونية) والمغناطيسية التى تستخدم فى الاقراص والأشرطة ، والذاكرة الضوئية .

7. **الخصائص الفيزيائية** : الخصائص الفيزيائية المستخدمة فى تخزين البيانات مهمة . ففي الذاكرة المتطايرة المعلومات تضمحل بشكل طبيعي أو يتم فقدانها عندما تفقد الذاكرة الطاقة الكهربائية . وفي الذاكرة الغير متطايرة (المستقرة) المعلومات المسجلة لا تُفقد أو تضمحل حتى يتم تغييرها بشكل متعمد ولا تحتاج إلى الطاقة الكهربائية للحفاظ على المعلومات . الذاكرة المغناطيسية غير متطايرة ، أما ذاكرة أشباه الموصلات (الألكترونية) فقد تكون متطايرة أو غير متطايرة .

8. **التنظيم** : فى ذاكرة الوصول العشوائي التنظيم هو قضية تصميمية والمقصود بالتنظيم هنا كيفية الترتيب الفعلي للخانات لتشكل كلمات (ترتيب شرائح الذاكرة للحصول على كلمة) .

7.1.2 التسلسل الهرمي للذاكرة

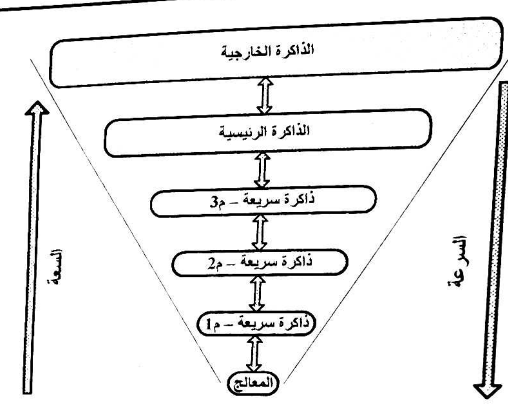
يمكن تلخيص المعوقات فى تصميم ذاكرة الحاسب فى ثلاثة أسئلة : كم السعة؟ ، كم السرعة؟ ، كم التكلفة؟. مسألة السعة هى الى حد ما مفتوحة ، فكلما زادت السعة تطورت التطبيقات البرمجية لكى تستغلها . أما السرعة فهى قضية مباشرة،

ولتحقيق أكبر قدر من الأداء يجب أن تكون الذاكرة قادرة على مواكبة سرعة المعالج بمعنى كلما كان المعالج ينفذ التعليمات لا نريد أن يكون لديه وقفة أنتظار للحصول على تعليمات أو معاملات . واخيراً ، السؤال الأخير ينبغي النظر فيه، للحصول على نظام عملي يجب أن تكون تكلفة الذاكرة معقولة بالنسبة للمكونات الأخرى . وكما كان متوقعاً ، فهناك مفاضلة بين الخصائص الرئيسية الثلاث للذاكرة وهي : السعة وزمن الوصول (السرعة) و التكلفة.

وتستخدم تقنيات متنوعة لتصميم و إنجاز نظم الذاكرة ، ومن خلال هذا الطيف من التقنيات تبرز العلاقات التالية :

- أسرع زمن وصول ، أكثر تكلفة لكل خانة .
- أكثر سعة ، أقل تكلفة لكل خانة .
- أكثر سعة ، أبطأ زمن وصول .

المعضلة التي تواجه المصمم هى واضحة ، فالمصمم يرغب في إستخدام تقنيات ذاكرة تضمن سعة كبيرة للذاكرة نتيجة أن السعة الكبيرة مطلوبة وتكلفة كل خانة أقل ، ومع ذلك ، لتلبية متطلبات الأداء المصمم يحتاج إلى إستخدام ذاكرة مكلفة ونسبيا سعتها قليلة وزمن وصولها قصير ، والشكل (7.1) يوضح التسلسل الهرمي وعلاقة كل مستوى مع زمن الوصول و السعة .



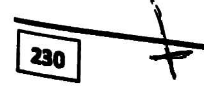
الشكل (7.1) - التسلسل الهرمي للذاكرة وعلاقته مع زمن الوصول والسعة

والسبيل للخروج من هذه المعضلة هو عدم الأعتماد على عنصر أو تقنية واحدة للذاكرة ولكن توظيف تسلسل هرمي للذاكرة . والشكل (7.2) يوضح التسلسل الهرمي النموذجي لنظام الذاكرة ، وكلما تنخفض في التسلسل الهرمي تلاحظ ما يلي :



1. تقل التكلفة لكل خانة .
2. تزداد السعة .
3. يزداد زمن الوصول
4. تنخفض وتيرة الوصول للذاكرة من قبل المعالج (تكرار التواصل).

وهكذا أصغر حجماً وأكثر تكلفة وأسرع تقابلها ذاكرة أكبر وأرخص وأبطأ .
ومفتاح النجاح لهذا التنظيم هو الفقرة (4) : تخفيض وتيرة الوصول للذاكرة .



المسجلات
الذاكرة السريعة
الذاكرة الرئيسية
ذاكرة محلية
القرص المغناطيسي
القرص الضوني
تخزين داخلى
الشريط المغناطيسى
تخزين خارجى

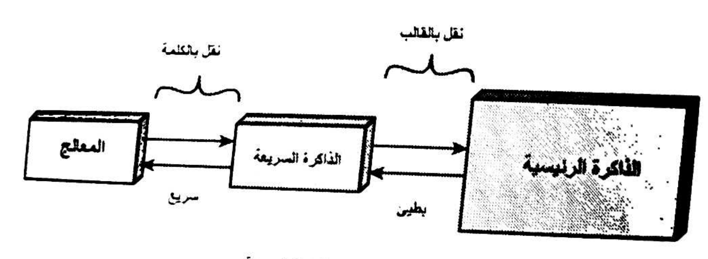
الشكل (7.2) - التسلسل الهرمي لنظام الذاكرة

7.2 مبادئ الذاكرة السريعة (Cache)

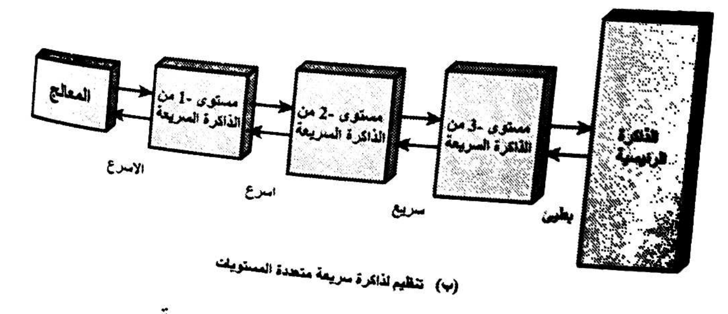
مبدأ الذاكرة السريعة يطمح لإعطاء سرعة للذاكرة تقترب من أسرع ذاكرة متوفرة ، وفي الوقت نفسه توفير حجم كبير للذاكرة بتكلفة أقل نوع من أنواع الذاكرة الألكترونية . ويتجلى هذا المفهوم في الشكل (7.3 – أ) ، فهناك ذاكرة رئيسية كبيرة نسبياً وبطيئة جنبا إلى جنب مع ذاكرة سريعة أصغر وأسرع ، والذاكرة السريعة تحتوي على نسخ من أجزاء في الذاكرة الرئيسية .

عندما يريد المعالج قراءة كلمة من الذاكرة ، فسيتم إجراء فحص لتحديد ما إذا كانت الكلمة في الذاكرة السريعة أم لا، إذا كان الأمر كذلك فسيتم تسليم الكلمة إلى المعالج . وإن لم يكن ، فسيتم قراءة قالب من الذاكرة الرئيسية والذى يتكون من عدد محدد من الكلمات إلى الذاكرة السريعة ومن ثم يتم تسليم الكلمة إلى

المعالج . والشكل (7.3 – ب) يوضح استخدام مستويات متعددة من الذاكرة السريعة ، فالذاكرة السريعة في المستوى-2 أبطأ وأكبر عادة من الذاكرة السريعة في المستوى-1 ، والذاكرة السريعة في المستوى-3 أبطأ وأكبر عادة من الذاكرة السريعة في المستوى-2 .




(أ) مستوى واحد من الذاكرة السريعة

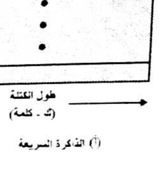


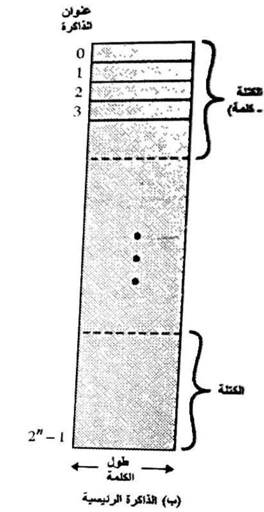
(ب) تنظيم لذاكرة سريعة متعددة المستويات

الشكل (7.3) – الذاكرة الرئيسية والسريعة

الشكل (7.4) يصور بنية نظام الذاكرة الرئيسة والذاكرة السريعة . فالذاكرة الرئيسية تتكون الى ما يصل ن-2 عنوان كلمة ، وكل كلمة لها عنوان وحيد بعرض س من الخانات . ولغرض المطابقة مابين الذاكرة الرئيسة والسريعة ، فإن الذاكرة الرئيسة تتألف من عدد من القوالب بطول ثابت (القالب عبارة عن عدد محدود من المواقع) ، وطول القالب (ن) من الكلمات ، وبذلك يكون بالذاكرة الرئيسة م = ن/ ن-2 قالب . والذاكرة السريعة تتكون من عدد (ص) قالب ، وتسمى **سطور** (السطر بسعة قالب) ، وكل منها يحتوي على (ن) كلمة بالإضافة الى علامة بخانات محدودة . وكل سطر يتضمن أيضا خانة للتحكم (لا تظهر) وذلك لبيان ما إذا كان تم تعديل السطر منذ أن تم تحميله أم لا .

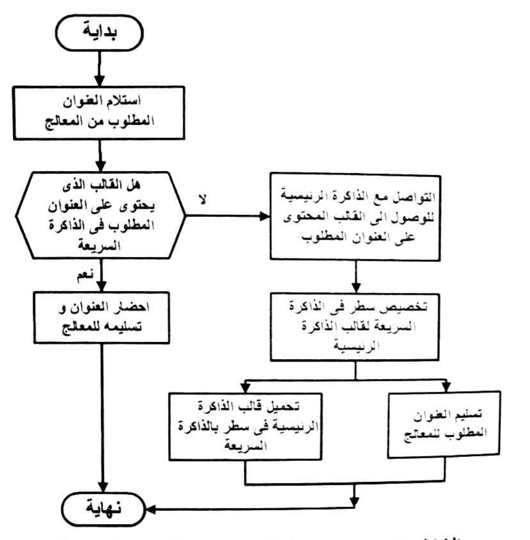
عدد السطور هو أقل بكثير من عدد قوالب الذاكرة الرئيسة (ص << م) ، وفي أي لحظة ، فإن عدد محدود من قوالب الذاكرة موجود في سطور الذاكرة السريعة . فإذا قرأت كلمة بقالب في الذاكرة الرئيسة يتم نقل ذلك القالب إلى أحد سطور الذاكرة السريعة ، وبما ان القوالب أكثر بكثير من السطور لا يمكن لسطر معين ان يكون بشكل وحيد ودائم مخصص لقالب ما ، ولذلك توجد علامة بالسطر تبين أى قالب من الذاكرة الرئيسية مخزن به حاليا . والعلامة هى فى العادة جزء من عنوان بالذاكرة الرئيسية . والشكل (7.5) يوضح عملية القراءة ، فالمعالج ينتج عنوان قراءة (العنوان المطلوب - RA) لقرأة كلمة واحدة ، فإذا كانت الكلمة في الذاكرة السريعة يتم تسليمها إلى المعالج . وخلاف ذلك ، يتم تحميل القالب الذى يحتوي على تلك الكلمة إلى الذاكرة السريعة ومن تم يتم تسليم الكلمة إلى المعالج ، والشكل (7.5) يوضح أن العمليتين الأخيرتين تحدثان بالتوازى في نفس الوقت ويعكس هذا التنظيم الشكل (7.6) ، وهو الأمر المعهود فى تنظيم الذاكرة السريعة المعاصرة .



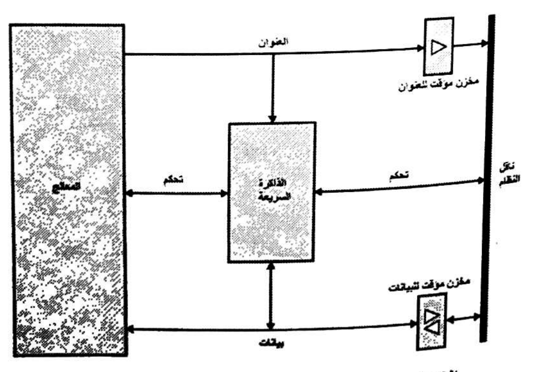


الشكل (7.4) - بنية نظام الذاكرة الرئيسية والذاكرة السريعة

في هذه التنظيم ، الذاكرة السريعة تتصل بالمعالج عن طريق خطوط البيانات **والتحكم و العنوان** . خطوط البيانات مرتبطة بمخزن مؤقت للبيانات و خطوط **العنوان مرتبطة** بمخزن مؤقت للعنوان ، المخزنين مرتبطين بناقل النظام الذي **يتصل بالذاكرة الرئيسية** . عند القراءة من الذاكرة السريعة يُعطل المخزن مؤقت **للبيانات والمخزن مؤقت للعنوان** (الشكل - 7.6) و يتم الأتصال فقط بين المعالج **والذاكرة السريعة** . عند عدم توفر البيانات فى الذاكرة السريعة يتم تحميل العنوان **المطلوب على نظام الناقل** ويتم أسترجاع البيانات من خلال المخزن المؤقت **البيانات إلى الذاكرة السريعة والمعالج فى نفس الوقت** .



الشكل (7.5) – عملية القراءة من الذاكرة السريعة



الشكل (7.6) – التنظيم النموذجى للذاكرة السريعة

7.3 عناصر تصميم الذاكرة السريعة

هذا القسم يقدم لمحة عامة عن معايير تصميم الذاكرة السريعة وتقرير بعض النتائج النموذجية ، ويضم ما يلي :

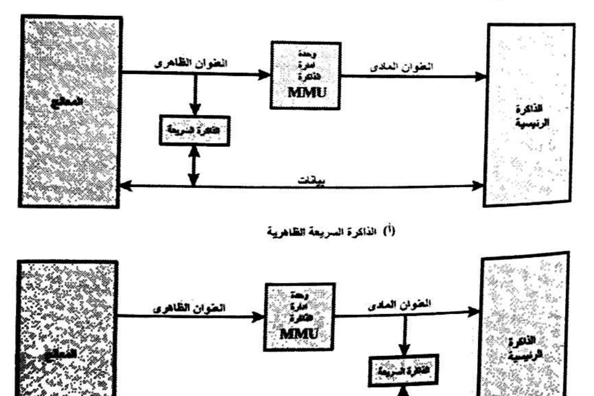
- العنونة : ظاهرية/إفتراضية أو مادية .
- الحجم .
- طريقة المطابقة : مباشرة أو ترابطية .
- خوارزميات الأستبدال :
  - أقل أستخدام مؤخراً (LRU)
  - الأول دخول أول خروج (FIFO)
  - الأقل تكرار إستخداماً (LFU)
  - عشوائياً .
- سياسة الكتابة :
  - الكتابة الذاتية .
  - إعادة الكتابة .
- حجم الكتلة
- مستويات الذاكرة السريعة : مستوى واحد أو أثنين ، و موحدة أم منفصلة.

7.3.1 عنونة الذاكرة السريعة

أغلب المعالجات المعاصرة تدعم أستخدام الذاكرة الظاهرية/الأفتراضية ، فالذاكرة الافتراضية خاصية تسمح للبرامج بالعنونة للذاكرة من ناحية أفتراضية بدون الأخذ في الأعتبار الحجم المادى المتوفر من الذاكرة الرئيسية ، وعند أستخدام العنوان في الذاكرة الأفتراضية ، فإن حقول العناوين الموجودة فى

التعليمة تحتوى على عناوين افتراضية ، وللقراءة أو للكتابة من الذاكرة الرئيسية تقوم وحدة ادارة الذاكرة (MMU) بترجمة العنوان الأفتراضى الى عنوان مادى حقيقي فى الذاكرة الرئيسية .

عند أستخدام الذاكرة الأفتراضية يختار المصمم أن يضع الذاكرة السريعة مابين المعالج و وحدة ادارة الذاكرة أو بين وحدة أدارة الذاكرة والذاكرة الرئيسية وكما هو موضح في الشكل (7.7) . ففي **الذاكرة السريعة الظاهرية** تُخزن البيانات بأستخدام العناوين الأفتراضية ، والمعالج يصل إلى الذاكرة السريعة مباشرة دون المرور عبر وحدة إدارة الذاكرة . و**الذاكرة السريعة المادية** تُخزن البيانات بأستخدام عناوين الذاكرة الرئيسية المادية الحقيقية.



(أ) الذاكرة السريعة الظاهرية

(ب) الذاكرة السريعة المادية

الشكل (7.7) – الذاكرة السريعة الظاهرية والمادية

7.3.2 حجم الذاكرة السريعة

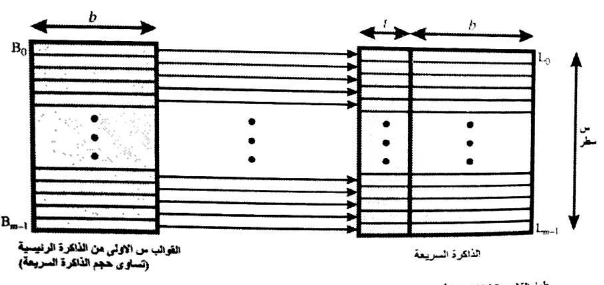
نود أن يكون حجم الذاكرة السريعة صغيراً بما فيه الكفاية بحيث يكون متوسط التكلفة الإجمالية للخانة بالذاكرة الداخلية بالكامل هو قريب منه فى الذاكرة الرئيسية فقط (تكلفة الذاكرة السريعة ضئيلة جدا مقارنة مع الرئيسية) ، وفى نفس الوقت نريد أن يكون حجمها كبيراً بما فيه الكفاية بحيث يكون أجمالى متوسط زمن الوصول قريبا من الذاكرة السريعة لوحدها (كبر حجم الذاكرة السريعة يقلل من التواصل مع الذاكرة الرئيسية – البيانات المطلوبة من المعالج متوفرة فى الذاكرة السريعة – مما يزيد من سرعة الذاكرة عامةً) ، وهناك دوافع أخرى عديدة للتقليل من حجم الذاكرة السريعة منها أن كبر الحجم قد يقلل من السرعة وكذلك المساحة المتاحة على رقيقة المعالج محدودة .

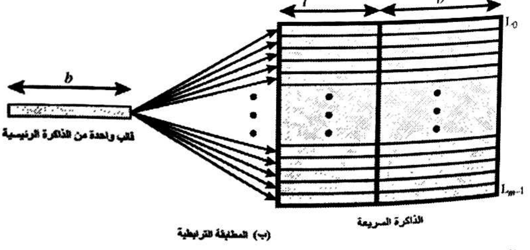
مما سبق يتضح أن قضية الحجم تخضع لمفاضلة مابين التكلفة و الأداء ، ولذلك كان هناك تنوع فى أنظمة الحاسب بناءً على ذلك ، فبعض الأنظمة تركز على الأداء بغض النظر عن التكلفة والأخرى بالعكس وذلك حسب المستهدف من تصميم نظام الحاسب .

7.3.3 طريقة المطابقة

لأن عدد سطور الذاكرة السريعة أقل بكثير جداً من عدد قوالب الذاكرة الرئيسية فكانت الحاجة إلى طريقة لتحميل (مطابقة) قوالب الذاكرة الرئيسية علي سطور الذاكرة السريعة ، وعلاوة على ذلك كانت هناك حاجة إلى وسيلة لتحديد أى قالب من الذاكرة الرئيسية يشغل حاليا سطر فى الذاكرة السريعة . وهناك طريقتين أساسيتين لتنظيم المطابقة هما كالتالى :

المطابقة المباشرة : وهى أبسط تقنية معروفة ، **وفيها يتم تحميل كل قالب من** الذاكرة الرئيسية في سطر محدد من الذاكرة السريعة ، **وكما هو موضح فى الشكل** (7.8 – أ) . والعيب الرئيسي هو أن هناك مكان مخصص ثابت لكل قالب معين، وبالتالي ، إذا كان البرنامج يؤشر الى كلمات مراراً وتكراراً فى قالبين مختلفين لكن يشغلان نفس السطر فإنه سيتم تبادل القالبين باستمرار في الذاكرة السريعة بغض النظر عن أحتمال وجود سطور أخرى شاغرة في الذاكرة السريعة .





الشكل (7.8) – مطابقة الذاكرة الرئيسية علي الذاكرة السريعة : المباشرة والترابطية

المطابقة الترابطية : وبهذه الطريقة يتم التغلب على عيوب المطابقة المباشرة من خلال السماح لكل قالب بالذاكرة الرئيسية بأن يتم تحميله في أي سطر بالذاكرة السريعة كما هو موضح بالشكل (7.8– ب) .

7.4.3 خوارزميات الأستبدال

عند تعبئة الذاكرة السريعة وجلب قالب جديد ، فلابد من إستبدال أحد السطور الموجودة . وفى المطابقة المباشرة هناك سطر محدد لكل قالب معين ، وليس هناك أى خيار آخر فى الأستبدال . ولكن فى المطابقة الترابطية هناك حاجة لخوارزمية للأستبدال ، وهناك عدد من الخوارزميات وسنذكر منها الأكثر شيوعا .

الأقل أستخداماً مؤخراً (LRU) : يستعاض عن ذلك القالب في المجموعة ألتي في الذاكرة السريعة أطول مدة مع عدم الإشارة إليها (عدم التواصل مع محتوياتها). والأحتمال الآخر هو الأول دخول أول خروج (FIFO) : ويستعاض عن ذلك القالب في المجموعة التي أمضت أطول مدة فى الذاكرة السريعة (أول دخولاً) . وآخر إحتمال هو الأقل وتيرة استخداماً (LFU) : ويستعاض عن ذلك القالب في المجموعة التي شهدت أقل عدد مرات من القراءة .

7.4.4 سياسة الكتابة

عندما يكون قالب من الذاكرة الرئيسية موجوداً في الذاكرة السريعة ووجب إستبداله هناك حالتين يجب أخذهما فى الاعتبار ، فإذا كان القالب قديماً ولم يتم تعديله يمكن كتابة قالب جديد عليه من دون الحاجة لإعادة كتابته أولاً فى الذاكرة الرئيسية ، أما إذا تم تعديل ولو كلمة واحدة من ذلك السطر فى الذاكرة السريعة فيجب أن يتم تحديث الذاكرة الرئيسية من خلال كتابة ذلك السطر من الذاكرة

السريعة إلى قالبه فى الذاكرة قبل أن يكتب قالب جديد . وهناك مجموعة متنوعة من سياسات الكتابة ، ويطلق على أبسطها تقنية **"الكتابة الذاتية"** ، وبأستخدام هذه التقنية ، فإنه يتم إجراء كافة عمليات الكتابة إلى الذاكرة الرئيسية وكذلك إلى الذاكرة السريعة فى نفس الوقت (بالتوازى) لضمان أن الذاكرة الرئيسية دائما صحيحة ويمكن لأي معالج أخر فى النظام مراقبة حركة المرور إلى الذاكرة الرئيسية للحفاظ على التناسق مع ذاكرته السريعة (يعدل محتوى ذاكرته حسب التعديل الذى يطرأ على الذاكرة الرئيسية) ، والعيب الرئيسي لهذا الأسلوب هو أنه ينتج عنه حركة مرور كبيرة . والتقنية البديلة تعرف باسم **"إعادة الكتابة"** للتقليل من الكتابة للذاكرة الرئيسية ، ففى هذه التقنية يتم إجراء التحديثات فقط في الذاكرة السريعة ومن ثم يعاد تحديث الذاكرة الرئيسية إذا تم تعديل نسختها فى الذاكرة السريعة وذلك عن طريق فحص خانة علامة السطر فى الذاكرة السريعة والتى تتغير إذا تم تعديل محتوى السطر .

7.4.5 حجم القالب

هناك عنصر آخر فى التصميم هو حجم القالب/السطر . فعندما يتم إسترداد قالب من البيانات ووضعه في الذاكرة السريعة ، ففي هذه الحالة يتم إسترداد الكلمة المطلوبة وبعض الكلمات الملاصقة لها ، وكلما زاد حجم القالب إلى حجم أكبر يتم جلب مزيد من البيانات المفيدة الى الذاكرة السريعة ولكن إحتمال إستخدام البيانات التى تم إحضارها حديثا يصبح أقل من إحتمال إعادة إستخدام البيانات التى تم استبدالها . وعلى العموم هناك إعتباران يجب أخذهما فى الأعتبار فى هذه الحالة :

- **قالب أكبر** يخفض من عدد القوالب التى يمكن أن تدرج في الذاكرة السريعة .
- كلما كبر القالب أصبحت أى كلمة أضافية بعيدة عن الكلمة المطلوبة .

7.4.6 مستويات الذاكرة السريعة

مع بداية ظهور الذاكرة السريعة كان لها مستوى واحد ، ولكن مع تطور التقنية **أصبح أستخدام** ذاكرة سريعة بمستويات متعددة أمر طبيعى . ومع تعدد المستويات **أصبح من** الممكن أن يكون المستوى الأول ملاصق للمعالج على الشريحة نفسها **(ذاكرة سريعة داخلية)** والمستوى الثانى على اللوحة (ذاكرة سريعة خارجية) .

هناك أستراتيجيتان فى تصميم الذاكرة السريعة ، فهي موحدة أو منفصلة . **فالذاكرة الموحدة** تحتوى على البيانات والتعليمات معاً بمعنى أنهما يشتركان فى **نفس الذاكرة السريعة** ، أما المنفصلة فذاكرة سريعة للتعليمات وآخرى للبيانات.

من مميزات الذاكرة السريعة الموحدة أن إحتمال وجود البيانات المطلوبة عالى **مقارنة بالمنفصلة** نتيجة لأنها توازن مابين جلب التعليمات وجلب البيانات تلقائياً، **وكذلك أقل تعقيداً** فى تصميم نظام الذاكرة لأنه سيتم تصميم ذاكرة واحدة فقط **للبيانات والتعليمات** . وميزة الذاكرة السريعة المنفصلة أنها تحد من التزاحم مابين **البيانات والتعليمات** فى عملية الجلب وهذا مهم جدا فى تصميم المعالجات ذات **المعالجة التواردية للتعليمات** إذ جلب التعليمات منفصل عن جلب البيانات .

مصطلحات مهمة

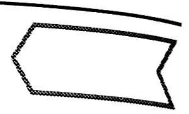
موقع	Location
وصول	Access
النوع المادى	Physical Type
متطاير	Volatile
زمن الوصول	Access Time
التاخير	Latency
علامة	Tag
قالب	Block
سريع	Fast
بطئ	Slow
سطر	Line
الذاكرة السريعة الأفتراضية	Virtual Cache
الذاكرة السريعة الموحدة	Unified Cache
وحدة إدارة الذاكرة	Memory Management Unit (MMU)
الكتابة الذاتية	Write Through
المطابقة الترابطية	Associative Mapping
المطابقة المباشرة	Direct Mapping
الذاكرة الظاهرية	Logical Cache
هيكلية الذاكرة	Memory Hierarchy
الذاكرة السريعة المتعددة المستويات	Multilevel Cache
الذاكرة السريعة المادية	Physical Cache
الوصول العشوائي	Random Access
خوارزمية الأستبدال	Replacement Algorithm
الذاكرة السريعة	Cache Memory
الوصول المتسلسل	Sequential Access
الذاكرة السريعة المنفصلة	Split Cache
إعادة الكتابة	Write Back
عنوان قراءة	Read Address (RA)
القرص الضوئي	Optical Disc (CD, DVD, …)
القرص المغناطيسي	Megantic Disc
مطابقة تناظرية	Mapping
الأول دخولاً أول خروجاً	First In First Out (FIFO)
الأقل أستخدماً مؤخراً	Least Recently Used (LRU)

الأقل وتيرة إستخدما	Least Frequently Used LFU)
خانة ثنائية	Bit
ثمان – 8 خانات ثنائية	Byte

أسئلة للمراجعة

1. ماهى العلاقة العامة مابين زمن الوصول والتكلفة والسعة للنظام الهرمى للذاكرة ؟
2. ماهو الفرق مابين المطابقة المباشرة والمطابقة الترابطية فى الذاكرة السريعة ؟
3. ناقش المخطط الأنسيابى لعمل الذاكرة السريعة؟
4. ناقش خوارزميات الأستبدال فى الذاكرة السريعة ومتى تستعمل؟
5. ماهى الخصائص التى تصنف على ضوئها الذاكرة؟
6. ماهو الفرق بين الوصول المتسلسل والوصول المباشر والوصول العشوائي؟
7. ناقش المبدأ العام لأختيار حجم الذاكرة السريعة ؟
8. ماهى معايير تصميم الذاكرة السريعة ؟
9. أفترض ذاكرة رئيسية من 32 قالب (من 0 إلى 31) وذاكرة سريعة من 8 أسطر (0 إلى 7) ، باستخدام المطابقة المباشرة ، ماهو القالب من الذاكرة الرئيسية الموجودة فى السطر 2 من الذاكرة السريعة ؟
10. أفترض ذاكرة رئيسية من 36 قالب (من 0 الى 35) وذاكرة سريعة من 6 أسطر (0 إلى 5) ، بإستخدام المطابقة الترابطية ، ما هى القوالب من الذاكرة الرئيسية الموجودة فى الذاكرة السريعة ، علما بان تسلسل التأشير أو المرجعية إلى الذاكرة الرئيسية هى كما يلى و طريقة الأستبدال فى الذاكرة السريعة هى (FIFO) :

ترتيب التأشير	1	2	3	4	5	6	7	8	9	10	11	12
الكتلة المؤشرة لها	2	23	15	23	24	16	15	17	18	30	17	23



الفصل الثامن

الذاكرة الداخلية

8 - الذاكرة الداخلية

نبدأ هذا الفصل مع دراسة مسحية للنظم الفرعية للذاكرة الرئيسية الإلكترونية (أشباه الموصلات) ، وتتضمن أنواع الذاكرة : ذاكرة القراءة فقط (ROM) والذاكرة التفاعلية (DRAM) والذاكرة الساكنة (SRAM) ، ومن ثم نتطرق إلى تقنيات التحكم فى الأخطاء والمستخدمة فى تعزيز موثوقية الذاكرة ، وفي النهاية نتطرق إلى البنية المتقدمة للذاكرة التفاعلية .

8.1 الذاكرة الرئيسية الألكترونية

في أنظمة الحاسب الأولى كانت ذاكرة الوصول العشوائي الرئيسية فى التخزين بالحاسب والأكثر شيوعاً هى أستخدام مصفوفة من الحلقات المغناطيسية على شكل كعكي ويشار إليها بالقوالب ، ولذلك ، فغالبا ما يشار إلي الذاكرة الرئيسية بالقالب، وهو مصطلح لا يزال قائما حتى يومنا هذا . ومع التطور والمزايا الكثيرة لتقنية الإلكترونيات الدقيقة (أشباه الموصلات) أصبح أستخدام الرقائق الإلكترونية كذاكرة رئيسية عالمياً وتغلبت على الذاكرة المغناطيسية ، وسيتم إستكشاف الجوانب الأساسية للذاكرة الألكترونية في هذا الجزء .

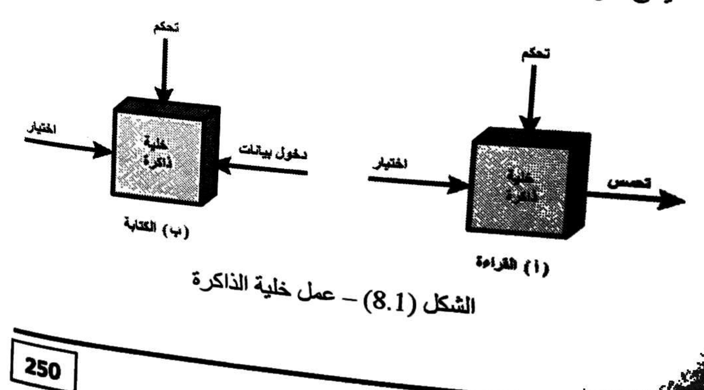
8.1.1 التنظيم الاساسي

العنصر الأساسي فى الذاكرة الالكترونية (أشباه الموصلات) **هى خلية الذاكرة .**
وعلى الرغم من أستخدام مجموعة متنوعة من التقنيات الإلكترونية ولكن كل خلايا الذاكرة الإلكترونية تشترك فى بعض الخصائص :

- **تُظهر حالتين مستقرتين** (أو شبه مستقرة) ، بحيث يمكن إستخدامهما لتمثيل **الثنائي 1 و 0** .
- **قابلية للكتابة** (على الأقل مرة واحدة) بتعريف حالة الخلية (1 أو 0).
- **قابلية للقراءة** بتحسس حالة الخلية (1 أو 0).

يوضح الشكل (8.1) عمل خلية الذاكرة . والشائع فإن الخلية لديها ثلاث منافذ **وظيفية قادرة** على حمل الإشارة الكهربائية : منفذ الإختيار بحيث يتم إختيار خلية **الذاكرة لعملية** القراءة أو الكتابة ، ومنفذ التحكم يُحدد نوع العملية قراءة أم كتابة، **وفي الكتابة يُدخل** المنفذ الثالث الإشارة الكهربائية التي تحدد حالة الخلية بقيمة 1 **أو 0 ، أما في القراءة** فيتم إستخدام المنفذ الثالث لإخراج/إظهار حالة الخلية (1 أو **0)**.

تفاصيل التنظيم الداخلي والأداء وتزامن خلية ذاكرة يعتمد على تقنية الدوائر **المتكاملة المستخدمة** . ولغرض الشرح سوف نُسلم بأن الخلايا يمكن إختيارها **لعملياتي القراءة والكتابة** .



الشكل (8.1) – عمل خلية الذاكرة

8.1.2 أنواع الذاكرة الألكترونية

جميع أنواع الذاكرة التي ستُبحث في هذا الفصل هي ذات وصول عشوائي ، حيث يتم الوصول المباشر لكل كلمة فردية في الذاكرة من خلال عنوانها وذلك بواسطة منطق عنونة معين (منطق مادى) . والجدول (8.1) يسرد الأنواع الرئيسية للذاكرة الإلكترونية (أشباه الموصلات) ، ويشار إليها عامة بذاكرة الوصول العشوائي (RAM) ، وهذا بطبيعة الحال إستخدام غير صحيح للمصطلح لأن كل الأنواع المدرجة في الجدول هي ذات وصول عشوائي . ومن الخصائص المميزة لذاكرة الوصول العشوائي هى أنه من الممكن قراءة بيانات من الذاكرة وكتابة بيانات جديدة إلى الذاكرة على حد سواء بسهولة وبسرعة ، كذلك يتم إنجاز كل من القراءة والكتابة من خلال إستخدام إشارات كهربائية.

الجدول (8.1) – أنواع الذاكرة الإلكترونية (أشباه الموصلات)

نوع الذاكرة	التصنيف	المسح	آلية الكتابة	أستقرار البيانات
ذاكرة الوصول العشوائي (RAM)	ذاكرة قراءة – كتابة	كهربائى ، بمستوى كلمة (8 خانات)	كهربائية	متطايرة
ذاكرة القراءة فقط (ROM)	ذاكرة قراءة فقط	غير ممكن	تصنيعى	مستقرة
ذاكرة القراءة فقط القابلة للبرمجة (PROM)			كهربائية	
ذاكرة القراءة فقط القابلة للمسح و البرمجة (EPROM)	ذاكرة قراءة – غالباً	أشعة فوق البنفسجية ، بمستوى الشريحة		
ذاكرة القراءة فقط القابلة للمسح و البرمجة كهربائيا (EEPROM)		كهربائى ، بمستوى كلمة (8 خانات)		
الذاكرة الوميضية (Flash Memory)		كهربائى ، بمستوى قالب		

السمة الأخرى المميزة لذاكرة الوصول العشوائي هو أنها متطايرة ، ولذلك يجب **توفير إمدادات** طاقة مستمرة لذاكرة الوصول العشوائي ، فإذا قطعت إمدادات **الطاقة** ستُفقد البيانات ، وبالتالي يمكن إستخدام ذاكرة الوصول العشوائي في **التخزين** المؤقت فقط .

النوعان التقليديان لذاكرة الوصول العشوائي والمستخدمة في أجهزة الحاسب هي **الذاكرة** التفاعلية (DRAM) والذاكرة الساكنة (SRAM) .

8.1.3 الذاكرة التفاعلية (DRAM)

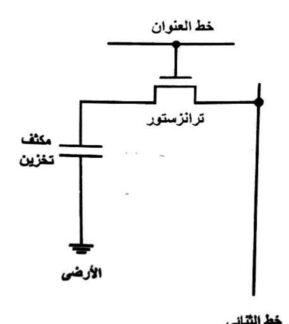
تنقسم تقنية ذاكرة الوصول العشوائى (RAM) الى قسمين إثنين : ساكنة وتفاعلية . فذاكرة الوصول العشوائى التفاعلية (DRAM) تُصنع من خلايا تخزن البيانات كشحنة كهربائية على المكثفات ، و وجود أو غياب الشحنة على المكثف يفسر كثنائى 1 أو 0 . ولأن المكثفات لديها ميل طبيعي لفقد الشحنة ، لذلك ذاكرة الوصول **العشوائى التفاعلية** تتطلب شحن تنشيطي دوري للمحافظة على البيانات المخزنة ، **ومصطلح** تفاعلية يشير إلى هذا الإتجاه حيث أن الشحنة المخزنة لها ميل للتسرب **حتى** مع الإستمرار فى التغذية بالطاقة .

الشكل (8.2 – أ) يوضح البنية النموذجية للذاكرة التفاعلية مكونة من خلية واحدة **تخزن خانة ثنائية واحدة** (Bit) . ويتم تنشيط خط العنوان عندما يراد قراءة أو كتابة **قيمة خانة هذه الخلية** . الترانزستور يعمل بمثابة المفتاح ويُغلق (يسمح للتيار **بالمرور**) إذا تم تطبيق جهد كهربائى على خط العنوان ويُفتح (لا يمر التيار) إذا لم **يطبق جهد على خط العنوان** .

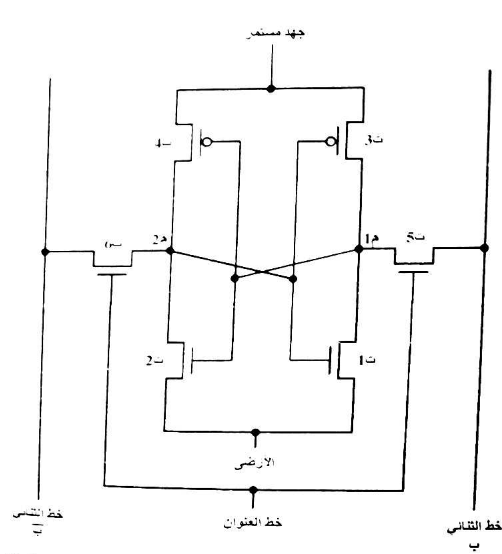
في عملية الكتابة يتم تطبيق إشارة جهد على خط الخانة ؛ **والجهد العالي يمثل** المنطق 1 ويمثل الجهد المنخفض المنطق 0 ، ومن ثم يتم **تطبيق إشارة على خط** العنوان (إختيار الخلية) الذى يسمح بإنتقال الشحنة إلى المكثف .

في عملية القراءة وعند إختيار خط العنوان ، فإن الترانزستور **يشتغل وتنتقل** الشحنة المخزنة على المكثف للخارج بواسطة خط الخانة إلى مضخم حساس **والذى** بدوره يقارن شحنة المكثف بقيمة مرجعية ويحدد ما إذا كانت الخلية **تحتوى على** منطق 0 او منطق 1 . والقراءة من الخلية تفرغ المكثف لذلك **يجب إعادة تخزينها** لإستكمال عملية القراءة .

وعلى الرغم من إستخدام خلية الذاكرة التفاعلية لتخزين **قيمة خانة ثنائية واحدة** (0 أو1) فهى في الأساس جهاز تناظرى . فالمكثف **يمكن أن يخزن أي قيمة ضمن** نطاق محدد ؛ وقيمة جهد العتبة يحدد ما إذا كان يتم تفسير هذه **الشحنة كمنطق** 1 أو 0 (تقارن كمية الشحنة مع قيمة جهد العتبة فأذا زادت تفسر 1 **وإذا نقصت تفسر** 0) .



الشكل (8.2 – أ) – **البنية النموذجية لخلية الذاكرة – خلية تفاعلية (DRAM)**



الشكل (8.2 – ب) – البنية النموذجية لخلية الذاكرة - خلية ساكنة (SRAM)

8.1.4 الذاكرة الساكنة (SRAM)

ذاكرة الوصول العشوائى الساكنة (SRAM) هى نظام رقمي يستخدم نفس عناصر المنطق المستخدمة في المعالج . ففي الذاكرة الساكنة يتم تخزين القيم الثنائية باستخدام بوابات منطق القلابات التقليدية ، والذاكرة الساكنة تحتفظ ببياناتها طالما تزويدها بالطاقة مستمر .

الشكل (8.2 – ب) يوضح البنية النموذجية لخلية من الذاكرة الساكنة . ودائرة **الخلية توضح أربعة الترانزستورات** (ت1، ت2، ت3، ت4) متصلة على شكل **متقاطع لتنتج حالة بمنطق مستقر** . ففي الحالة المنطقية 1 ، فإن النقطة م1 عالية

ونقطة م2 منخفضة وفي هذه الحالة ، ت1 و ت4 لا تعمل ، و ت2 و ت3 تعمل . في الحالة المنطقية 0 ، النقطة م1 منخفضة ونقطة م2 عالية وفي هذه الحالة ، ت1 و ت4 تعمل ، و ت2 و ت3 لا تعمل . وكلتا الحالتين مستقرة طالما طبق جهد تيار مستمر (DC) ، وخلافاً للذاكرة التفاعلية لا حاجة للتنشيط للاحتفاظ بالبيانات .

كما في الذاكرة التفاعلية فإنه يتم إستخدام خط العنوان الخاص بالذاكرة الساكنة لفتح أو إغلاق المفتاح . وخط العنوان يتحكم بالترانزستورات (ت5 و ت6) ، وعندما يتم تطبيق إشارة على هذا الخط يفتح الترانزستورين وذلك للسماح بعملية القراءة أو الكتابة . لعملية الكتابة فإنه يتم تطبيق القيمة المرجوة للخانة على الخط **ب** ، ويتم تطبيق المكمل لها (المعكوس) على الخط ب̄ ، وهذا يفرض على الترانزستورات الأربعة (ت1، ت2، ت3، ت4) أن تكون في حالة صحيحة . ولعملية القراءة تُقرأ قيمة الخانة من الخط **ب** .

عند المقارنة بين الذاكرة الساكنة والذاكرة التفاعلية فإن الذاكرتين الساكنة والتفاعلية متطايرة بمعنى يجب توفير الطاقة بشكل مستمر للذاكرة للحفاظ على قيم الخانة ، وخلية الذاكرة التفاعلية هى أبسط وأصغر من الذاكرة الساكنة ، ولذلك التفاعلية أكثر كثافة (خلايا أصغر - خلايا أكثر على نفس وحدة مساحة) وأقل تكلفة من خلايا الذاكرة الساكنة المناظرة لها ، ومن ناحية أخرى تتطلب الذاكرة التفاعلية دوائر تنشيط داعمة .

فى الذاكرة ذات السعة الكبيرة فإن التكلفة الثابتة لدوائر التنشيط تعوض بالتكلفة الصغيرة المتغيرة لخلايا الذاكرة التفاعلية ، ولذلك فالذاكرة التفاعلية تميل إلى أن تكون المفضلة لمتطلبات الذاكرة الكبيرة . وثمة نقطة أخيرة هي أن الذاكرة الساكنة عموماً نوعا ما أسرع من الذاكرة التفاعلية ، وبسبب هذه الخصائص النسبية يتم

إستخدام الذاكرة الساكنة فى ذاكرة التخزين المؤقت – الذاكرة السريعة – داخل وخارج الشريحة ، وتستخدم الذاكرة التفاعلية فى الذاكرة الرئيسية لنظام الحاسب.

8.1.5 ذاكرة القراءة فقط (ROM)

وكما يوحي أسمها ، فذاكرة القراءة فقط (ROM) تحفظ نمط دائم من البيانات بشكل لا يمكن تغييرها . وذاكرة القراءة فقط غير متطايرة ؛ بمعنى أنها لا تتطلب مصدر دائم للطاقة للحفاظ على قيم الخانة في الذاكرة . وفي حين أنه من الممكن القراءة من ذاكرة القراءة فقط ، فإنه ليس من الممكن كتابة بيانات جديدة فيها . ذاكرة القراءة فقط تستخدم فى تطبيق مهم وهو البرمجة الدقيقة ، والتطبيقات الأخرى المحتملة تشمل ما يلي :

- مكتبة بالبرامج الفرعية للوظائف المطلوبة بإستمرار .
- برامج النظام
- جداول الدوال .

ميزة ذاكرة القراءة فقط هو أن البيانات أو البرنامج موجود بشكل دائم في الذاكرة الرئيسية (جزء من الذاكرة الرئيسية عبارة عن ذاكرة قراءة فقط) ولا تحتاج أبداً لأن يتم تحميلها ثانية من جهاز التخزين الثانوي . ويتم تصنيع ذاكرة القراءة فقط مثل أي شريحة دوائر متكاملة أخرى ، ولكن مع تثبيت البيانات سلكياً على الشريحة كجزء من عملية التصنيع . وهذا يعرض مشكلتين : خطوة إدراج البيانات أثناء خطوات عملية التصنيع تتضمن تكلفة ثابتة كبيرة نسبيا ،وذلك سواء أنتجت واحدة أو آلاف النسخ من ذاكرة القراءة فقط ، وثانيا ، ليس هناك مجال للخطأ ، فخطأ فى خانة واحدة عند التصنيع ينتج عنه رمي المجموعة بأكملها من شرائح ذاكرة القراءة فقط المصنعة .

عند الحاجة لعدد قليل من ذاكرة القراءة فقط بمحتوي معين وُجد بديلاً أقل تكلفة وهو ذاكرة القراءة فقط القابلة للبرمجة : (PROM) . ومثل ذاكرة القراءة فقط فإن ذاكرة القراءة فقط القابلة للبرمجة (PROM) غير متطايرة وربما قابلة للكتابة مرة واحدة فقط (أو عدة مرات) . وفى ذاكرة القراءة فقط القابلة للبرمجة يتم تنفيذ عملية الكتابة كهربائيا ويمكن إجرائها من قبل المورد أو العميل في وقت لاحق بعد تصنيع الشريحة الأصلي ، ولعملية الكتابة أو "البرمجة" تتطلب معدات خاصة . ذاكرة القراءة فقط القابلة للبرمجة توفر مرونة وملاءمة ولكن تبقى ذاكرة القراءة فقط (ROM) جذابة لحاجات الإنتاج بكميات كبيرة .

نتيجة التطبيقات المعاصرة وتطور تقنيات التصنيع تم أنتاج ذاكرة القراءة – غالبا وهى تطوير عن ذاكرة القراءة فقط ، وهى مفيدة للتطبيقات التي تكون بها عمليات القراءة أكثر تواترا من عمليات الكتابة ولكن تتطلب تخزين غير متطاير (مستقر)، وهناك ثلاثة أشكال شيوعاً من ذاكرة القراءة – غالبا : ذاكرة القراءة فقط القابلة للمسح والبرمجة (EPROM) وذاكرة القراءة فقط القابلة للمسح والبرمجة كهربائيا (EEPROM) ، والذاكرة الوميضية (فلاش) .

ذاكرة القراءة فقط القابلة للمسح والبرمجة (EPROM) تتم القراءة و الكتابة بها كهربائيا وكما هو الحال مع ذاكرة القراءة فقط القابلة للبرمجة (PROM) ، ولكن قبل عملية الكتابة تتم عملية مسح ضوئي للشريحة وذلك لمسح جميع خلايا التخزين لتعود للحالة الأولية عن طريق تعريض الشريحة المعبأة للأشعة فوق البنفسجية . **ويتم تنفيذ المسح من خلال تسليط ضوء فوق البنفسجي مكثف من خلال نافذة تم تصميمها في شريحة الذاكرة لهذا الغرض ، ويمكن تنفيذ عملية المسح مراراً وتكراراً ؛ وتنفيذ كل عملية مسح يمكن أن يصل إلى 20 دقيقة . وهكذا ، يمكن**

تعديل ذاكرة القراءة فقط القابلة للمسح والبرمجة عدة مرات ومثل ذاكرة القراءة **فقط وذاكرة القراءة فقط القابلة للبرمجة** يمكن لذاكرة القراءة فقط القابلة للمسح **والبرمجة حفظ البيانات إلى ما لا نهاية تقريبا** ولكميات متساوية من التخزين فإن **ذاكرة القراءة فقط القابلة للمسح والبرمجة** هي أكثر تكلفة من ذاكرة القراءة فقط **القابلة للبرمجة ولكن لديها ميزة القدرة على التحديث المتكرر لمحتوياتها.**

هناك شكل أكثر جاذبية من أنواع ذاكرة القراءة – غالبا هى ذاكرة القراءة فقط **القابلة للمسح** و البرمجة كهربائيا (EEPROM) ذاكرة القراءة فقط القابلة للمسح **والبرمجة** كهربائيا يمكن الكتابة بها في أي وقت دون الحاجة لمسح محتوياتها **السابقة ويمكنها** تحديث كلمة (ثمان) او عدة كلمات . عملية الكتابة تأخذ وقتا **أطول بكثير** من عملية القراءة (عدة مئات من الميكروثانية لكل كلمة) . ذاكرة **القراءة فقط القابلة للمسح والبرمجة كهربائيا** (EEPROM) تجمع بين الإستفادة **من كونها غير متطايرة** مع مرونة كونها قابلة للتحديث في مكانها وذلك بإستخدام **الخطوط العادية لناقل التحكم** والعناوين والبيانات . وذاكرة القراءة فقط القابلة **للمسح والبرمجة كهربائيا** (EEPROM) هى أكثر تكلفة من ذاكرة القراءة فقط **القابلة للمسح والبرمجة** (EPROM) وأيضا أقل كثافة ، وبعدد خانات أقل لكل **شريحة .**

شكل آخر من أشكال الذاكرة الالكترونية (أشباه الموصلات) هي الذاكرة الوميضية (Flash Memory) (أطلق عليها هذا الأسم بسبب السرعة التي يمكن برمجتها)، **وقدمت لأول مرة في منتصف** 1980 . الذاكرة الوميضية هي وسيطة بين ذاكرة **القراءة فقط القابلة للمسح والبرمجة** وذاكرة القراءة فقط القابلة للمسح والبرمجة **كهربائيا فى التكلفة والوظيفة** . ومثل ذاكرة القراءة فقط القابلة للمسح والبرمجة

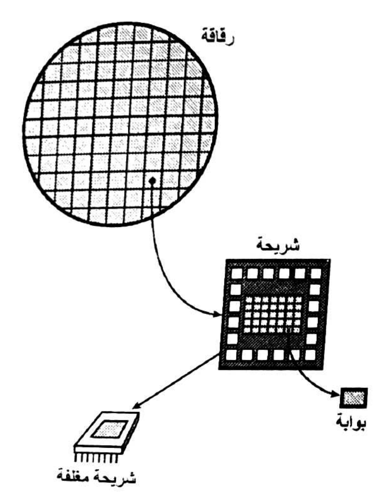
كهربائيا فإن الذاكرة الوميضية تستخدم تقنية المسح الكهربائي **ويمكن أن تمسح** الذاكرة الوميضية بالكامل في ثانية أو بضع ثوان وهذا أسرع بكثير من ذاكرة القراءة فقط القابلة للمسح والبرمجة ، بالإضافة إلى ذلك يمكن **مسح قوالب معينة** من الذاكرة بدلا من الشريحة بأكملها .

الذاكرة الوميضية حصلت على إسمها لأن شريحة الذاكرة منظمة بحيث يمكن مسح قسم من خلايا الذاكرة في اجراء واحد أو "ومضة"، ولكن الذاكرة الوميضية لا توفر مسح على مستوى الكلمات (وحدة المسح قالب) . ومثل ذاكرة القراءة فقط القابلة للمسح والبرمجة فإن الذاكرة الوميضية تستخدم ترانزستور واحد فقط لكل خانة ولذلك تحقق كثافة عالية مثل ذاكرة القراءة فقط القابلة للمسح والبرمجة .

8.1.6 شرائح الذاكرة الالكترونية

الذاكرة الإلكترونية تأتي في شرائح مغلفة كما هو موضح فى الشكل (8.3) ، وكل شريحة تحتوي على مجموعة من خلايا الذاكرة . وفي التسلسل الهرمي للذاكرة ككل هناك مفاضلة بين السرعة والقدرة والتكلفة ، وهذه المفاضلة موجودة أيضاً عند النظر إلى تنظيم خلايا الذاكرة والمنطق الوظيفي للشريحة .

فى الذاكرة الإلكترونية فإن إحدى القضايا الرئيسية فى التصميم هو عدد خانات البيانات التي يمكن قراءتها/كتابتها في آن واحد . وأحد الحلول هو **تنظيمى وفيها** تنظم الذاكرة بحيث يكون الترتيب الفعلي للخلايا فى المصفوفات هو **نفس الترتيب** المنطقي للكلمات فى الذاكرة (كما يُنظر إليه من قبل المعالج) .



الشكل (8.3) – العلاقة بين الرقاقة و الشريحة والبوابة المنطقية

يتم تنظيم مصفوفة الذاكرة من (س) من الكلمات لكل منهما (ص) خانة (الذاكرة يمكن اعتبارها كمصفوفة ذات بعدين ، عدد الكلمات×عرض كل كلمة) .

فعلى سبيل المثال فإنه يمكن تنظيم شريحة ذاكرة بسعة 16- ميغا خانة على شكل 1- ميغا كلمة ذات 16- خانة لكل كلمة (السعة : 1- ميغا كلمة × 16- خانة لكل كلمة = 16- ميغا خانة) ، وهو ما يسمى تنظيم خانة –لكل– شريحة ، وحيث تتم قراءة/كتابة بيانات خانة في كل مرة . والشكل (8.4) يبين التنظيم النموذجي لذاكرة تفاعلية (DRAM) بسعة 16- ميغا خانة ، وفي هذه الحالة تتم قراءة أو كتابة 4- خانات في المرة الواحدة ، ومنطقيا مصفوفة الذاكرة منظمة فى أربع مصفوفات مربعة كل منها بعدد (2048) فى (2048) عنصر .

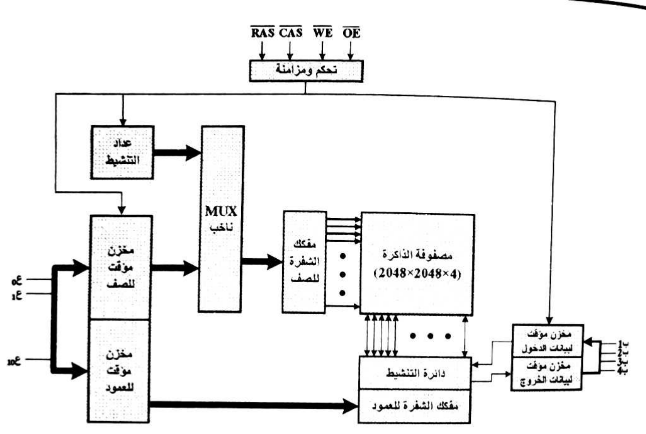
إن حجم معين من الذاكرة يمكن تنظيمه بعدة ترتيبات مادية مختلفة ، **وفي كل** ترتيب فإن عناصر المصفوفة ترتبط بخطوط أفقية (صف) ورأسية (عمود) ، وكل خط أفقي يرتبط بمنفذ الإختيار لكل خلية في هذا الصف ، وكل خط عمودي يرتبط بمنفذ دخول البيانات/التحسس فى كل خلية من هذا العمود . وخطوط العنوان تقدم عنوان الكلمة التي سيتم اختيارها ، ونحتاج هنا إلى ما مجموعه $\log_2$ س من الخطوط ، وحيث س هو حجم الذاكرة بعدد الكلمات و ليس عرضها (الذاكرة يمكن اعتبارها كمصفوفة ذات بعدين ، عدد الكلمات×عرض كل كلمة). في مثالنا المذكور سالفاً ، كانت هناك حاجة إلى (11) خط عنوان لإختيار صف من (2048) صف ($2^{11} = 2048$) ، وهذه (11) خط يتم تغذيتها الى وحدة فك الترميز بالصف الذى له (11) خط دخول و (2048) خط خروج ، ومنطق وحدة فك الترميز ينشط أحدى هذه (2048) مخرج إعتماداً على نمط الخانة (القيمة الثنائية) الموجود على خطوط الدخول (11) . ونحتاج إلي (11) خط عنوان إضافى لإختيار واحد من (2048) عمود حيث كل عمود متصل 4 - خانات ، وتستخدم أربعة خطوط بيانات وذلك من أجل إدخال وإخراج 4 - خانات من و إلى مخزن مؤقت للبيانات . ففى حالة الإدخال (الكتابة) يتم تنشيط مشغل الخانة الخاص بكل خط خانة الى 1 أو 0 حسب قيمة خط البيانات المناظر ، وفى حالة الإخراج (القراءة) يتم تمرير قيمة كل خط خانة من خلال حساس مضخم إشارة وتوضع القيمة على خط البيانات . وخط الصف يحدد أى صف من الخلايا الذى يستخدم للقراءة أو للكتابة .

نتيجة أن بيانات 4-خانات فقط يمكن قراءتها/كتابتها لهذه الذاكرة التفاعلية (DRAM) ، فيجب أن تكون هناك عدة شرائح من الذاكرة التفاعلية متصلة بوحدة

تحكم الذاكرة لقراءة/كتابة كلمة من البيانات الى الناقل (على إعتبار أن الكلمة تتكون **من 8 خانات**) .

فى الشكل الذى يوضح المثال (الشكل (8.4)) نلاحظ أن هناك (11) خط عنوان **فقط** (ع0 - ع10) وهو نصف العدد المتوقع لمصفوفة ذاكرة بحجم (2048×2048) **كلمة ، وذلك للتقليل** من عدد مسامير الإدخال/الإخراج الخارجية للشريحة . **والخطوط** (22) المطلوبة لعناوين هذه الذاكرة التفاعلية يتم تمريرها من خلال (4×4 **ميغا** $= 4 \times 2048 \times 2048 = 4 \times 2^{11} \times 2^{11}$ - 22 خط عنونة) **منطق إختيار** خارج الشريحة وتُختزل الى (11) خط عنوان (مشتركة) . فأولا ، **يتم تمرير الجزء الأول** من العنوان على (11) خط العنوان إلى الشريحة لتحديد **عنوان صف فى مصفوفة** الذاكرة ، ومن ثم يتم تمرير الجزء الأخير على (11) **خط العنوان لتحديد** العمود فى مصفوفة الذاكرة . وترافق هذه الإشارات (اشارات **العنوان على 11 خط**) إشارة إختيار عنوان الصف ($\overline{RAS}$) واشارة أختيار عنوان **العمود** ($\overline{CAS}$) **وذلك لتوفير** التزامن مع الشريحة ، ومسمار تفعيل الكتابة ($\overline{WE}$) **وتفعيل الخرج** ($\overline{OE}$) تُحدد ما إذا كانت العملية كتابة أو قراءة . مسمارين آخريين **ليسا موضحين في الشكل** (8.4) ، وهما الأرضى (V_{SS}) ومصدر الجهد (V_{CC}) .

ان أسلوب مشاركة العنوان وإستخدام مصفوفات مربعة نتج عنه مضاعفة فى حجم **الذاكرة الى أربعة أمثال مع كل جيل** جديد من شرائح الذاكرة .



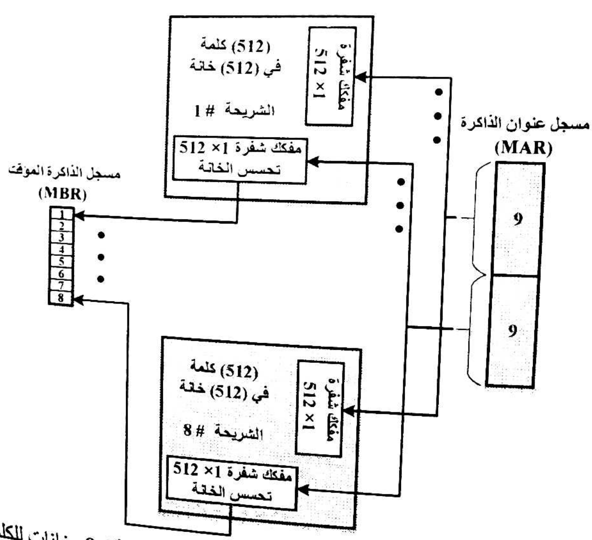
الشكل (8.4) - التنظيم النموذجي لذاكرة تفاعلية (DRAM) بسعة 16- ميغا خانة (4×4 ميغا)

الشكل (8.4) يبين أيضا إدراج دوائر التنشيط فى الذاكرة حيث أن جميع شرائح الذاكرة التفاعلية تتطلب عملية تنشيط ، وهناك تقنية بسيطة للتنشيط وهى **في الواقع** تعطل شريحة الذاكرة التفاعلية عن التواصل الخارجي أثناء تنشيط **البيانات فى كل** الخلايا ، ففي خلال عملية التنشيط تتم قراءة البيانات وإعادة كتابتها **في نفس الموقع** وبهذا يتم تنشيط كل خلية في الصف .

8.1.7 تنظيم وحداة الذاكرة الالكترونية

إذا كانت شريحة ذاكرة الوصول العشوائي تحتوي على خانة ثنائية لكل كلمة ، فمن الواضح أننا سنحتاج على الأقل لعدد من الشرائح مساو لعدد خانات الكلمة . وكمثال على ذلك الشكل (8.5) يبين كيف يمكن تنظيم وحدة ذاكرة تتكون من 256 كيلو كلمة / 8 – خانات للكلمة . ومن أجل 256 كيلو كلمة فإننا بحاجة إلى عنوان بعرض

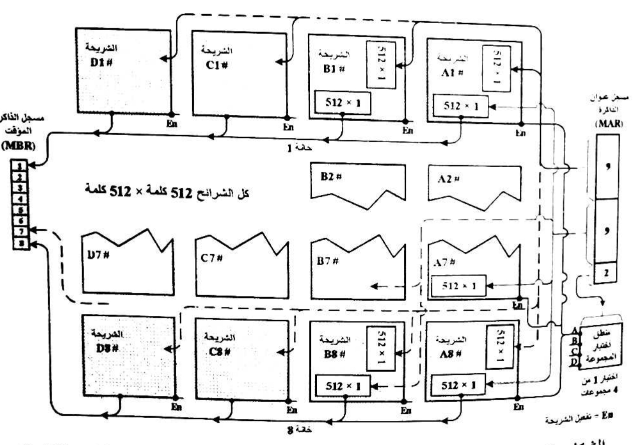
18- خانة (2^{18} = 256 كيلو) ويتم تزويده إلى الوحدة من مصدر خارجي (مثل **خطوط العنوان** من ناقل النظام المرفقة به الوحدة) . ويزود العنوان إلى 8 شرائح **بسعة 256 كيلو** كلمة / خانة للكلمة ، كل منها توفر إدخال/إخراج خانة . وهذا **التنظيم يعمل** طالما حجم الذاكرة يساوي عدد الخانات لكل شريحة ، وفي حالة الحاجة لذاكرة أكبر فإننا بحاجة لمجموعة أخرى من الشرائح .



الشكل (8.5) - تنظيم وحدة ذاكرة تتكون من 256 كيلو كلمة / 8–خانات للكلمة

الشكل (8.6) يبين كيفية تنظيم ذاكرة تتكون من 1 ميغا كلمة / 8 خانات لكل كلمة . **وفي هذه الحالة** لدينا أربعة أعمدة من الشرائح ، و رُتّب كل عمود بحيث **يحتوي 256 كيلو كلمة كما في الشكل** (8.5) . ومن أجل ذاكرة بسعة 1 ميغا كلمة

هناك حاجة إلى 20 خط عنوان ، ويتم توجيه 18 خانة الدنيا **للعنوان لكل وحدات** الذاكرة – 32 وحدة ، والخانتين العليتين من العنوان **(أول خانتين من اليسار) هي** مدخلات لوحدة منطق إختيار المجموعة التي تُرسل إشارة **تمكين شريحة إلى إحدى** وحدات الأعمدة الأربع (كل عمود مكون من 8 شرائح ذاكرة) .



الشكل (8.6) - تنظيم وحدة ذاكرة تتكون من 1 ميغا كلمة / 8- **خانات للكلمة**

8.2 الذاكرة المتداخلة (Interleaved)

تتكون الذاكرة الرئيسية من مجموعة شرائح ذاكرة تفاعلية ، **ويمكن تجميع عدد من** الشرائح معا لتشكيل صف ذاكرة . ومن الممكن **تنظيم صفوف الذاكرة بطريقة** تعرف باسم **الذاكرة المتداخلة** حيث أن كل صف **وبشكل مستقل قادر على خدمة** طلب من الذاكرة للقراءة أو الكتابة **بحيث أن نظام بعدد س من الصفوف يمكنه** تقديم س طلبات خدمة بالتوازي ، **ويتم التسريع من معدل نقل قالب البيانات من**

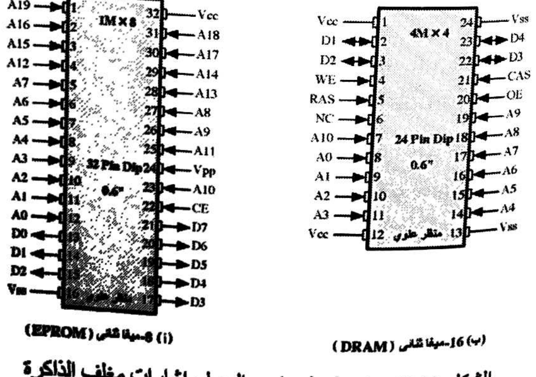
الذاكرة إذا تم تخزين الكلمات المتتالية في صفوف مختلفة (القالب مجموعة كلمات) من الذاكرة .

8.3 تغليف الشريحة

الدوائرة المتكاملة مثبتة فى غلاف/عبوة تحتوي على مسامير للربط الخارجي ، والشكل (8.7 – أ) يبين على سبيل المثال عبوة ذاكرة قراءة فقط القابلة للمسح والبرمجة (EPROM) ، وهى شريحة 8 – ميغا خانة نظمت على شكل 8 خانة × 1 – ميغا (8 خانات = بيانات ، 1 – ميغا = مواقع) . ففي هذه الحالة يتم التعامل مع هذا التنظيم كعبوة كلمة–واحدة–لكل–شريحة ، والعبوة تتضمن 32 مسمار خارجى ، وهي احدى أحجام الشرائح القياسية . مسامير الربط الخارجى تُعرف خطوط الإشارة التالية :-

- عنوان الكلمة المراد الوصول إليها ، ومن أجل 1 – ميغا كلمة (1 – ميغا = 2^{20}) مطلوب ما مجموعه 20 مسمار للعنوان (A_0-A_{19}) .
- البيانات المراد قراءتها (8 خانات) ، وتتكون من 8 – خطوط (D_0-D_7) .
- إمدادات الطاقة إلى الشريحة (V_{CC}) .
- مسمار التأريض (V_{SS}) .
- مسمار تمكين الشريحة (CE) . ونتيجة إحتمال أن يكون هناك أكثر من شريحة ذاكرة فى النظام ، وكل منها مرتبط بنفس ناقل العناوين ، فمسمار (CE) يستخدم لتحديد ما إذا كان العنوان هو لهذه الشريحة ام لا . ويتم تنشيط هذا المسمار بإستخدام منطق مرتبط بخانات الجزء الأعلى ترتيبا من ناقل العنوان المرتبط مع الشريحة (خانات العنوان مافوق A_{19}) .
- جهد البرمجة (Vpp) والذي يتم تفعيله أثناء البرمجة (عمليات الكتابة) .

تنظيم نموذجى لتوزيع المسامير الخارجية لذاكرة تفاعلية (DRAM) يوضحه الشكل (8.7 – ب) ، والشكل يوضح شريحة ذاكرة تفاعلية بسعة 16 – ميغا خانة منظمة على شكل 4 × 4 ميغا . وهناك عدة أختلافات عن شريحة ذاكرة القراءة فقط (EPROM) المبينة فى نفس الشكل ، ونذكر منها أنه نتيجة أنها شريحة ذاكرة تفاعلية فلذلك لها مسامير بيانات تستعمل فى إتجاهين (إدخال/إخراج) ، مسمار تفعيل الخرج (OE) و مسمار تفعيل الكتابة (WE) يوضحان أن العملية الجارى تنفيذها على الشريحة هى عملية قراءة أو كتابة ، وبما أن التواصل مع شريحة الذاكرة التفاعلية يتم عبر الصفوف و الأعمدة لذلك العنوان مشترك بحيث يتم تغذية الشريحة بعدد 11 مسمار خاصة بالعنوان لتحدد 4 ميغا تركيبة صف/عمود ($2^{11} \times 2^{11} = 2^{22} = 4$ ميغا) ، و وظيفة مسمار أختيار عنوان الصف (RAS) ومسمار أختيار عنوان العمود (CAS) هي تحديد ما أذا كان العنوان المدخل للصف أم للعمود .



(أ) 8-ميغا ثنائي (EPROM)

(ب) 16-ميغا ثنائي (DRAM)

الشكل (8.7) – نموذج لمسامير الربط وإشارات مغلف الذاكرة

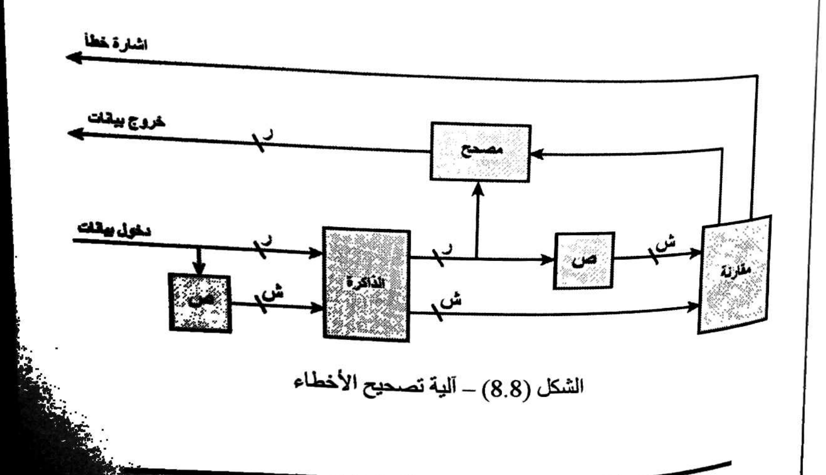
8.4 آلية تصحيح الأخطاء

نظام الذاكرة الإلكترونية معرض للأخطاء ، ويمكن تصنيف هذه الإخفاقات **بالفشل المادى أو الأخطاء اللينة** . الفشل المادى هو وجود خلل مادي دائم بحيث أن **الخلية أو الخلايا** المتضررة من الذاكرة لا يمكنها تخزين البيانات بشكل موثوق وتصبح **ثابتة في 0 أو 1** أو تتبدل بطريقة متقطعة مابين 0 و 1 ، ويمكن أن يكون سبب **الفشل** الإستخدام السيئ في ظروف بيئية قاسية أو عيوب فى التصنيع أو تلف مادى . **الأخطاء اللينة** هى عشوائية ، وحدث غير متلف يمكن أن يغير محتوي خلية ذاكرة **أو أكثر** دون الإضرار بالذاكرة ، ويمكن أن يكون سبب الأخطاء اللينة من مشاكل **إمدادات الطاقة أو جسيمات ألفا** ، وهذه الجسيمات تنتج عن الإضمحلال الإشعاعي **وذلك شائع** بسبب وجود نويّات مشعة بكميات صغيرة في كل المواد تقريبا . **الفشل المادى والأخطاء اللينة** غير مرغوب فيها مطلقا فى الذاكرة ، وأحدث أنظمة الذاكرة **الرئيسية** تشمل منطق لكشف وتصحيح الأخطاء ، الشكل (8.8) يوضح بعبارات **عامة** كيفية تنفيذ هذه العملية . فعند قراءة البيانات إلي الذاكرة يتم إجراء **عملية حسابية** عليها لإنتاج شفرة تحقق ، والمبينة في الدالة (ص) ، يتم تخزين كل من **شفرة التَحقق والبيانات** فى الذاكرة ، وبالتالي إذا كانت الكلمة التى ستخزن بطول **(ر)** خانة بيانات والشفرة المنتجة وهى بطول (ش) خانة ، إذن الحجم الفعلي للكلمة **المخزنة** هو (ر+ ش) خانة . وعند قراءة كلمة مخزنة سابقا يتم إستخدام الشفرة **لكشف وربما تصحيح الأخطاء** حيث يتم إنشاء قيمة جديدة للشفرة (ش) من خانات **البيانات المحفوظة** (ر) ومقارنتها مع خانات الشفرة المُحضرة (المحفوظة مع **البيانات**) ، والمقارنة تنتج احدى هذه النتائج :

- **لا توجد أخطاء ، ويتم إرسال خانات البيانات المُحضرة خارجاً .**

- تكتشف أخطاء ومن الممكن تصحيح الأخطاء ، **ويتم تغذية خانات** البيانات بالإضافة إلى خانات تصحيح الأخطاء **إلى وحدة التصحيح** والتى بدورها تنتج مجموعة من خانات مصححة (ر) **ليتم إرسالها** خارجاً .
- تكتشف أخطاء ولكن من غير الممكن **تصحيحها ويفاد تقرير بهذه** الحالة .

يشار إلى شفرات التحقق التي تعمل على هذا النحو **بشفرات تصحيح الأخطاء ،** وتُميز الشفرة بعدد خانات الأخطاء في الكلمة التى **يمكنها الكشف عنها وتصحيحها،** ومن أبسط و أشهر شفرات تصحيح الأخطاء هى شفرة **"هامنك" التي وضعها** "ريتشارد هامنك" في مختبرات "بيل" .



الشكل (8.8) – آلية تصحيح الأخطاء

8.5 التنظيم المتقدم للذاكرة التفاعلية

من أهم الأختناقات في نظام الحاسب عند أستخدام معالجات عالية الأداء هو الأرتباط مع الذاكرة الداخلية الرئيسية حيث يعتبر هذا الأرتباط هو الأكثر أهمية في نظام حاسب بأكمله .

لا تزال شريحة الذاكرة التفاعلية لبنة البناء الأساسية فى الذاكرة الرئيسية حتى الأن؛ ومنذ أوائل 1970 وحتى وقت قريب لم يكن هناك أي تغيير كبير في بنية الذاكرة التفاعلية .

الشريحة التقليدية للذاكرة التفاعلية مُقيدة بمعماريتها الداخلية وطريقة إرتباطها بناقل المعالج الخاص بالذاكرة . وأحد الحلول لمشكلة أداء الذاكرة التفاعلية كذاكرة رئيسية هو إدراج مستوى أو أكثر من الذاكرة الساكنة العالية السرعة للتخزين السريع بين الذاكرة الرئيسية و المعالج ، ولكن الذاكرة الساكنة هي أكثر تكلفة بكثير من الذاكرة التفاعلية ، كذلك أن توسيع حجم ذاكرة التخزين السريع أبعد من نقطة معينة يأتى بنتائج عكسية . وفي السنوات الأخيرة تمت عدة تحسينات على البنية الأساسية للذاكرة التفاعلية وبعض هذه التحسينات هى الأن في الاسواق .

من أهم أنواع الذاكرة التفاعلية التي تهيمن حاليا على الأسواق هي الذاكرة التفاعلية المتزامنة (SDRAM) ، والذاكرة التفاعلية مزدوجة السرعة (DDR-DRAM) وذاكرة رمبوس التفاعلية (RDRAM) . والجدول (8.2) يبين مقارنة فى الأداء مابين الأنوع المذكورة من الذاكرة .

الجدول (8.2) – مقارنة لأداء بعض انواع الذاكرة التفاعلية (DRAM)

نوع الذاكرة	تردد النبضة (ميغا هيرتز)	معدل النقل (غيغا ثمان/ثانية)	زمن الوصول (نانو ثانية)	عدد المسامير
SDRAM	166	1.3	18	168
DDR	200	3.2	12.5	184
RDRAM	600	4.8	12	162

8.5.1 الذاكرة التفاعلية المتزامنة (SDRAM)

واحد من أشكال الذاكرة التفاعلية الأكثر استخداما وعلى نطاق واسع هى الذاكرة التفاعلية المتزامنة ويرمز لها (SDRAM) . وليست مثل الذاكرة التفاعلية التقليدية (غير متزامنة) ، فالذاكرة التفاعلية المتزامنة (SDRAM) تتبادل البيانات مع المعالج بالتزامن مع نبضة ساعة خارجية وتشتغل بالسرعة الكاملة لناقل المعالج/الذاكرة وبدون فرض حالات أنتظار . وفي الذاكرة التفاعلية النموذجية يقدم المعالج إلى الذاكرة العناوين وإشارات التحكم ، والتى تشير إلى أن مجموعة من البيانات في موقع معين في الذاكرة ينبغي أن تُقراء من إو تُكتب في الذاكرة التفاعلية، وبعد تأخير محدد – زمن الوصول – الذاكرة التفاعلية إما ان تكتب أو تقرأ البيانات . وأثناء تأخير زمن الوصول فإن الذاكرة التفاعلية تُجرى عدة وظائف داخلية مختلفة مثل مبادلة شحن خطوط الصفوف والأعمدة ، وإستشعار البيانات ، وتوجيه البيانات للخارج من خلال الخازن المؤقت ، إذن المعالج ببساطة يجب عليه أن ينتظر خلال هذا التأخير مما يبطئ من الأداء العام للنظام . مع الوصول المتزامن فإن الذاكرة التفاعلية المتزامنة (SDRAM) تنقل البيانات من وإلى

المعالج تحت سيطرة ساعة النظام . فالمعالج يصدر إشارت التحكم ومعلومات عن **العنوان والتي تُمسك** من قبل الذاكرة التفاعلية المتزامنة ، ثم تستجيب الذاكرة **التفاعلية المتزامنة** بعد عدد معين من الدورات الزمنية ، وفي هذا الوقت نفسه يمكن **للمعالج بأمان** أن ينجز مهام أخرى أثناء قيام الذاكرة التفاعلية المتزامنة بمعالجة **الطلب** (قبل الأستجابة) .

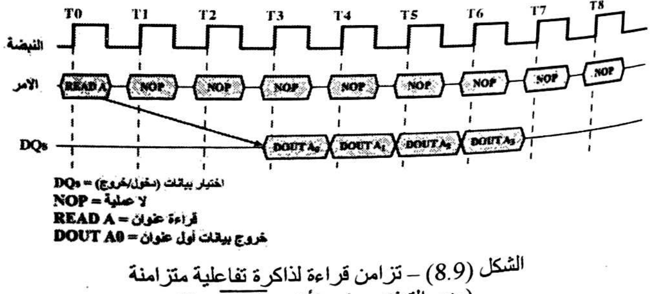
تستخدم الذاكرة التفاعلية المتزامنة وضع التدفق للحد من زمن إعداد العنوان وزمن **إعادة شحن** خط الصف والعمود بعد التواصل الأول . وفي وضع التدفق يمكن **لسلسلة** من خانات البيانات أن تُدفع للخارج على وجه السرعة بعد أن يتم الوصول **إلى الخانة الاولى** . وهذا الوضع مفيد عندما تكون جميع الخانات المراد الوصول **إليها في تسلسل** وفي نفس الصف من المصفوفة مثل الوصول الأول . بالإضافة **إلى ذلك** ، الذاكرة التفاعلية المتزامنة لديها بنية داخلية متعددة المصفوفات وذلك **يعمل على تحسين** فرص التوازي على الشريحة .

ميزة أخرى تتميز بها الذاكرة التفاعلية المتزامنة عن الذاكرة التفاعلية التقليدية هى **وضع المسجل** ومنطق التحكم المصاحب له ، وهو يوفر آلية لتعديل الذاكرة **التفاعلية المتزامنة** لتلائم إحتياجات نظام محدد . وضع المسجل يحدد مدى التدفق **(طول سلسلة البيانات)** ، وهو عدد الوحدات المنفصلة من البيانات التى ستتم تغذيتها **بشكل متزامن إلى الناقل** ، والمسجل يسمح للمبرمج بضبط التأخير مابين تلقي طلب **القراءة وبداية نقل البيانات** .

الأداء الأفضل للذاكرة التفاعلية المتزامنة يتم عند نقل قوالب كبيرة من البيانات بشكل تسلسلى لتطبيقات مثل معالجة النصوص وجداول البيانات والوسائط المتعددة . والشكل (8.9) يوضح مثال على عمل ذاكرة تفاعلية متزامنة ، وفي هذه

الحالة فإن مدى التدفق هو (4) والتأخير (2) . ويبدأ أمر القراءة المتدفق **بخفض** إشارتى ($\overline{CAS}$) و ($\overline{CS}$) ورفع إشارتى ($\overline{RAS}$) و ($\overline{WE}$) **مع حافة الإرتفاع فى** النبضة . والعنوان المُدخل يحدد عنوان عمود بداية التدفق ، **و وضع المسجل يحدد** نوع التدفق (تسلسلى أو متقطع) ومدى التدفق (1 ، 2 ، 4 ، 8 ، **صفحة كاملة)** ، والتأخير من بداية الإجراء وحتى ظهور بيانات الخلية الأولى **على المخارج يساوي** قيمة تأخير ($\overline{CAS}$) الذى تم تحديده في وضع المسجل .

مع تطور التقنية أنتجت حاليا نسخ محسنة من الذاكرة التفاعلية المتزامنة ، **وتُعرف** باسم الذاكرة التفاعلية المتزامنة مزدوجة السرعة ، **ويرمز لها اختصاراً** (DDR-SDRAM) . ذاكرة الذاكرة التفاعلية المتزامنة **مزدوجة السرعة مكنت** من التغلب على سقف تواصل واحد لكل دورة حيث **يمكنها إرسال البيانات إلى** معالج مرتين لكل دورة نبضة.



الشكل (8.9) – تزامن قراءة لذاكرة تفاعلية متزامنة
(مدى التدفق – 4 ، تأخير $\overline{CAS}$ – 2)

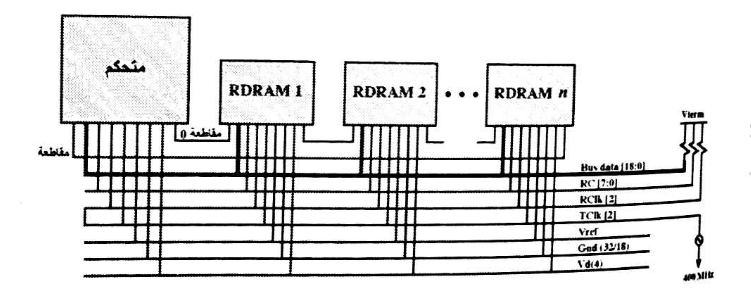
8.5.2 ذاكرة رامبوس التفاعلية (RDRAM)

تم تصميم ذاكرة رامبوس التفاعلية (RDRAM) من قبل شركة "رامبوس"، **وتم إستخدامها من قبل شركة آنتل فى معالجات البنتيوم وإيتانيوم**، وقد أصبحت **المنافس الرئيسي للذاكرة التفاعلية المتزامنة** (SDRAM). وشرائح ذاكرة **رامبوس التفاعلية مغلفة رأسيا وجميع مسامير الخرج على جانب واحد**. والشريحة **تتبادل البيانات مع المعالج عبر 28 سلك لا يتجاوز طولها 12 سنتيمتراً، ويمكن للناقل أن يتواصل مع عدد من شرائح ذاكرة رامبوس التفاعلية يصل إلى 320 شريحة، وبسرعة 1.6 غيغا ثُمان (GByte) في الثانية**، ويستخدم ناقل خاص **بذاكرة رامبوس التفاعلية يسلم العنوان وإشارات التحكم باستخدام بروتوكول خاص غير متزامن، وما يجعل هذه السرعة ممكنة هو تقنية الناقل نفسه** والتي تُعَرف **بدقة متناهية المعاوقات الكهربائية والنبضات والإشارات**. وبدلاً من أن يتحكم بها **مباشرة بواسطة إشارات RAS و CAS و W/R و CE كالمستخدمة في الذاكرة التفاعلية التقليدية**، تحصل ذاكرة رامبوس التفاعلية على طلب البيانات من ناقل **عالي السرعة ويحتوى الطلب على العنوان المطلوب، ونوع العملية، وعدد وحدات البيانات (الوحدة = 8 – خانات) في العملية**.

الشكل (8.10) يوضح مخطط لذاكرة رامبوس التفاعلية. والمخطط يتكون من **مُتَحكِم وعدداً من وحدات ذاكرة رامبوس التفاعلية متصلة عبر ناقل مشترك، بحيث أن المُتَحكِم موجود في الطرف الأول للناقل، وفى الطرف الأخير من الناقل توجد نهاية متوازية لخطوط الناقل**. والناقل يتضمن 18 خط بيانات (16 بيانات **فعلية، 2 تماثل)، ويعمل بدورة زمنية ضعف معدل دورة النبضة، بمعنى أنه يتم إرسال خانة عند حافة الإرتفاع وخانة عند حافة الإنخفاض لكل إشارة نبضة، وهذا**

ينتج عنه معدل إشارة لكل خط بيانات يصل الى 800 ميغا خانة في الثانية. وهناك **مجموعة منفصلة من الخطوط** – 8 خطوط (RC) تستخدم لإشارات العنوان **والتحكم، وهناك أيضا إشارة نبضية تُرسل من النهاية البعيدة عن المُتَحكِم ثم تنتشر وتعود بالعكس كحلقة**.

وحدة ذاكرة رامبوس التفاعلية ترسل البيانات إلى وحدة المُتَحكِم بالتزامن مع الساعة الرئيسية و وحدة المُتحكِم ترسل البيانات إلى ذاكرة رامبوس التفاعلية بالتزامن مع إشارة نبضة الساعة في الإتجاه العكسى، وخطوط الناقل المتبقية تشمل الجهد المرجعي والأرضى ومصدر الطاقة.



الشكل (8.10) – بنية ذاكرة رامبوس التفاعلية (RDRAM)

8.5.3 الذاكرة التفاعلية المتزامنة - مزدوجة السرعة (DDR SDRAM)

الذاكرة التفاعلية المتزامنة (SDRAM) محدودة **بحقيقة أنه يمكنها فقط إرسال البيانات إلى المعالج مرة واحدة لكل دورة نبضية للناقل**. والإصدار الجديد من **الذاكرة التفاعلية المتزامنة مزدوجة السرعة ويشار اليها (DDR-SDRAM) ذات معدل سرعة بيانات مُضاعف، حيث بإمكانها إرسال البيانات مرتين في كل**

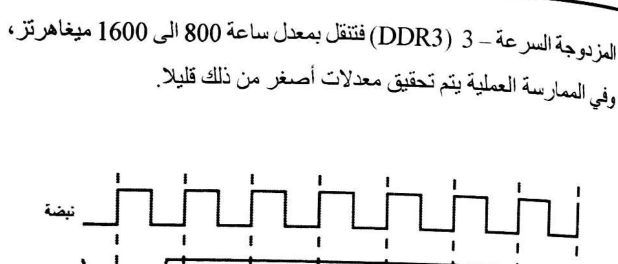
دورة نبضية ، فمرة عند حافة إرتفاع نبضة الساعة ومرة عند حافة الإنخفاض . ولقد وضعت الذاكرة التفاعلية المتزامنة مزدوجة السرعة من قبل مؤسسة (JEDEC) ، والأن العديد من الشركات تصنع الشرائح المزدوجة السرعة (DDR) ، وهى تستخدم على نطاق واسع في أجهزة الحاسب المكتبية والخوادم وغيرها من أنظمة الحاسب الآلى .

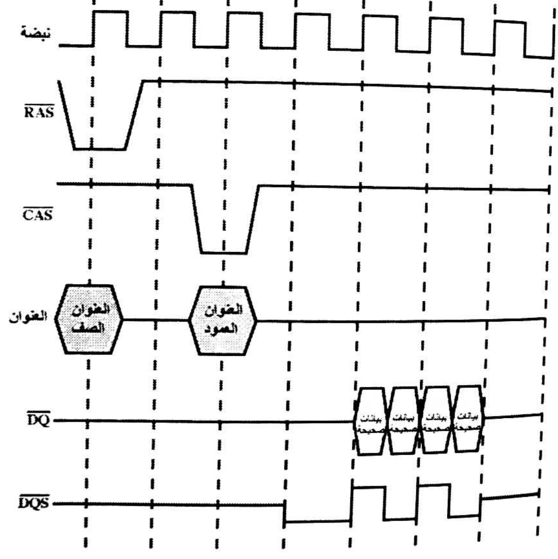
الشكل (8.11) يوضح التزامن الأساسي للقراءة فى الشرائح المزدوجة السرعة (DDR) . وتتم مزامنة نقل البيانات مع حافة الإرتفاع وحافة الإنخفاض للنبضة ، وتتم المزامنة أيضا مع إشعار البيانات ثنائي الإتجاه (DQS) الذى يُفعل من قبل وحدة تحكم الذاكرة أثناء القراءة ومن قبل الذاكرة التفاعلية أثناء الكتابة ، وفي التطبيقات النموذجية يتم تجاهل (DQS) أثناء القراءة .

يوجد جيلين من تقنية الشرائح المزدوجة السرعة المحسنة . فالشرائح المزدوجة السرعة – 2 (DDR2) زادت من معدل نقل البيانات عن طريق زيادة الوتيرة التشغيلية لشريحة ذاكرة الوصول العشوائي وزيادة المخزن المؤقت للجلب المُسبق من 2 إلى 4 خانة لكل شريحة ، والمخزن المؤقت للجلب المُسبق هو ذاكرة سريعة موجودة على شريحة ذاكرة الوصول العشوائي .

قدمت الشرائح المزدوجة السرعة – 3 (DDR3) في عام 2007 ، وزادت من حجم المخزن المؤقت للجلب المُسبق إلى 8 خانات . ونظريا يمكن لوحدة الشرائح المزدوجة السرعة (DDR) من نقل البيانات بمعدل نبضى في نطاق بين 200 و 600 ميغاهرتز ، أما وحدة الشرائح المزدوجة السرعة – 2 (DDR2) فيمكنها النقل بمعدل ساعة بين 400 إلى 1066 ميغاهرتز ، فى حين أن وحدة الشرائح

المزدوجة السرعة – 3 (DDR3) فتنقل بمعدل ساعة 800 الى 1600 ميغاهرتز، وفي الممارسة العملية يتم تحقيق معدلات أصغر من ذلك قليلا.





الشكل (8.11) – تزامن القراءة للذاكرة التفاعلية المتزامنة - مزدوجة السرعة (DDR SDRAM)

8.5.4 الذاكرة التفاعلية السريعة (CDRAM)

الذاكرة التفاعلية السريعة ويرمز لها (CDRAM) وضعتها شركة ميتسوبيشي ، حيث تُدمج ذاكرة ساكنة (SRAM) سريعة (16 كيلو خانة) مع شريحة ذاكرة تفاعلية (DRAM) . ويمكن إستخدام الذاكرة الساكنة مع الذاكرة التفاعلية السريعة (CDRAM) بطريقتين ، أولا ، يمكن استخدامها على انها ذاكرة سريعة حقيقية (Cashe Memory) وتتكون من خطوط بعدد 64 - خانة ، وهذا الوضع فعال للوصول العشوائي العادى إلى الذاكرة . ويمكن أيضا إستخدام الذاكرة الساكنة مع الذاكرة التفاعلية السريعة كمخزن موقت لدعم الوصول التسلسلى لقالب من البيانات، فعلى سبيل المثال ، لتحديث خانات بيانات شاشة يمكن للذاكرة التفاعلية السريعة أن تجلب مسبقا البيانات من الذاكرة التفاعلية الى مخزن موقت بتقنية الذاكرة الساكنة .

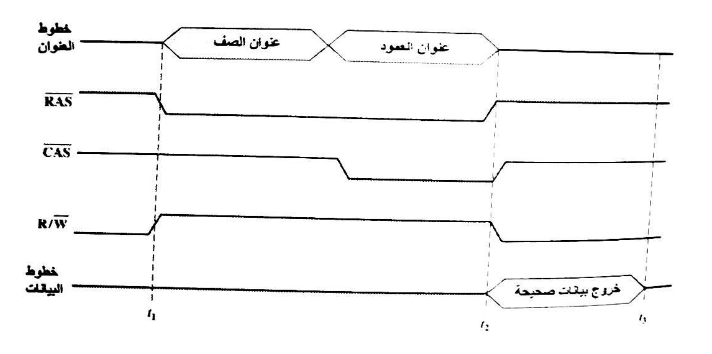
مصطلحات مهمة

ذاكرة القراءة - غالبا	Read-mostly memory
ذاكرة غير متطايرة (مستقرة)	Nonvolatile memory
ذاكرة القراءة فقط القابلة للبرمجة	Programmable ROM (PROM)
ذاكرة قراءة فقط	Read-only memory (ROM)
الذاكرة الساكنة	Static RAM (SRAM)
الذاكرة التفاعلية المتزامنة	Synchronous DRAM (SDRAM)
الخطأ الناعم	Soft error
ذاكرة متطايرة	Volatile memory
معدل بيانات مضاعف	Dual Data Rate - DDR
الذاكرة التفاعلية السريعة	Cache DRAM
الذاكرة التفاعلية	Dynamic RAM (DRAM)
ذاكرة القراءة فقط القابلة للمسح والبرمجة كهربائيا	Electrically erasable programmable ROM (EEPROM)
ذاكرة القراءة فقط القابلة للمسح والبرمجة	Erasable programmable ROM (EPROM)
شفرة تصحيح الاخطاء	Error correcting code (ECC)
الذاكرة الوميضية	Flash memory
ذاكرة رامبوس التفاعلية	RamBus DRAM (RDRAM)
الفشل المادى	Hard failure
ذاكرة الوصول العشوائى	Random Access Memory (RAM)
وضع التدفق	Burst Mode
تماثل	Parity
تفعيل الإخراج	Output Enable (OE)
تفعيل الكتابة	Write Enable (WE)
إختيار عنوان الصف	Raw Address Select (RAS)
إختيار عنوان العمود	Column Address Select (CAS)
خانة ثنائية	Bit
ناخب	Multiplexer (Mux)
ميغا (في النظام الثنائى)	Mega (2^{20})
غيغا (في النظام الثنائى)	Giga (2^{30})
كيلو (في النظام الثنائى)	Kilo (2^{10})
مفكك الشفرة	Decoder
ثمان - 8 خانات ثنائية	Byte

أسئلة للمراجعة

1. ماهو الفرق مابين أنواع الذاكرة التالية :
   أ- ذاكرة القراءة فقط القابلة للمسح والبرمجة .
   ب- ذاكرة القراءة فقط القابلة للمسح والبرمجة كهربائيا .
   ت- الذاكرة الوميضية .
2. ماهو الفرق بين الذاكرة الساكنة والتفاعلية بالنسبة الى السرعة والتكلفة والحجم و طريقة العمل ؟
3. ماهى آلية تصحيح الأخطاء فى الذاكرة الالكترونية؟
4. وضح التنظيم النموذجي لذاكرة تفاعلية بسعة 16 ميغا خانة (4M×4) باستخدام 11 خط عنونة ؟
5. بنأ على دراستك للذاكرة الداخلية ، وضح التنظيم النموذجي لذاكرة تفاعلية بسعة 32 ميغا خانة ثنائية (4M×8) باستخدام 11 خط عنونة ؟
6. الشكل (8.6) يوضح كيفية بناء وحدة ذاكرة بسعة تخزينية 1 ميغا ثُمان باستعمال مجموعة من 4 شرائح بسعة 256 كيلو ثُمان . ولنفترض أن هذه الشرائح مجمعة كاشريحة واحدة بسعة 1 ميغا ثُمان بعرض 8 خانات ثنائية للكلمة . وضح بالشكل التخطيطى لكيفية بناء ذاكرة بسعة 8 ميغا ثُمان لنظام حاسوبى بإستخدام ثمانية شرائح بسعة 1 ميغا ثُمان ، تأكد من توضيح خطوط العناوين في مخططك و لماذا هي مستعملة ؟
7. أفترض ذاكرة تفاعلية لها دورة تنشيط تبلغ 64 مرة لكل ملي ثانية و كل عملية تنشيط تتطلب 150 نانو ثانية في حين ان دورة الذاكرة تتطلب 250 نانو ثانية . ماهى نسبة زمن التنشيط لإجمالى زمن عمل الذاكرة ؟
8. **الشكل التالى يوضح توقيتات عملية قراءة لذاكرة تفاعلية على ناقل ، ومع أفتراض أن زمن التواصل يستغرق من t_1 إلى t_2 ، وكذلك يوجد زمن تنشيط يستغرق من t_2 إلى t_3 وفيه تنشط الذاكرة نفسها قبل أن تسمح للمعالج بالتواصل معها مرة اخرى . ومع أفتراض أن زمن التواصل هو 60 نانو ثانية وزمن التنشيط 40 نانو ثانية ، فماهو زمن دورة الذاكرة ، وماهو أقصي معدل بيانات يمكن أن توفره هذه الذاكرة مع أفتراض :**
   أ- ان خرج البيانات من الذاكرة خانة واحدة فقط (1-bit) .

ب- أن خرج البيانات من الذاكرة 32 – خانة ثنائية .



الفصل التاسع

الذاكرة الخارجية

9 - الذاكرة الخارجية

يتناول هذا الفصل الجزء الأخير من هيكلية نظام الذاكرة وهو الذاكرة الخارجية حيث سنتطرق إلى مجموعة من أجهزة وأنظمة الذاكرة الخارجية ، والبداية ستكون مع القرص المغناطيسي حيث تعتبر الأقراص المغناطيسية – أفتراضيا تقريبا – أساس الذاكرة الخارجية فى أنظمة الحاسب الآلى . وفى هذا الفصل سنتناول أيضاً أستخدام صفوف القرص المغناطيسي لتحقيق أداء عالى ونبحث على وجه التحديد تقنية نظام الصفوف المتكررة من الأقراص المستقلة والتي تُعرف أختصاراً **(RAID - Redundant Array of Independent Disks)** . الذاكرة الضوئية الخارجية ذات أهمية متزايدة ومكونة للعديد من أنظمة الحاسب وسيتم تناولها في القسم الثالث من هذا الفصل ، وأخيراً وصف للشريط المغناطيسي .

9.1 القرص المغناطيسي

القرص عبارة عن طبق دائري مصنوع من مواد غير مغنطيسية ويسمى الركيزة ويغلف بمواد قابلة للمغنطة . وتقليديا فالركيزة مصنوعة من الألمنيوم أو سبائك الألمنيوم ، وفي الأونة الأخيرة أدخلت ركائز الزجاج والركيزة الزجاجية لديها عدة مزايا ، منها ما يلي :

- **تحسين في إستواء سطح الغشاء المغناطيسي مما يزيد من موثوقية القرص.**
- **إنخفاض كبير في العيوب الكلية للسطح مما يساعد في تقليل أخطاء القراءة والكتابة.**

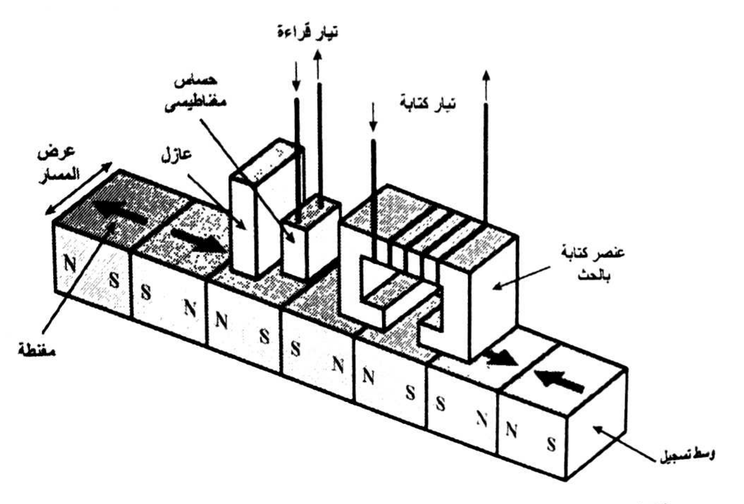
- القدرة على دعم أرتفاع منخفض لحركة طيران الرأس .
- صلابة أفضل وذلك بالتقليل من كمية الحركة التي يتعرض لها القرص .
- قدرة أكبر على تحمل الصدمات والأضرار .

9.1.1 الآلية المغناطيسية للقراءة والكتابة

يتم تسجيل البيانات وإسترجاعها لاحقا من على القرص بواسطة ملف توصيل **يدعى الرأس** ، وفي كثير من النظم يوجد رأسين إحدهما للقراءة وآخر للكتابة . وخلال عملية القراءة أو الكتابة فإن الرأس ثابت بينما يدور الطبق تحته .

آلية الكتابة تستغل حقيقة أن الكهرباء التي تتدفق في الملف ينتج عنها مجال **مغناطيسي** ، فلذلك يتم إرسال نبضات كهربائية في رأس الكتابة مما ينتج عنها **أنماط** مغناطيسية تسجل على السطح الذى تحت الرأس ، والتيار الموجب ينتج **أنماط** مختلفة عن التيار السالب . رأس الكتابة مصنوع من مادة قابلة للمغنطة بسهولة وهو على شكل كعكة مستطيلة مع وجود فجوة على طول جانب واحد وعليها القليل من الأسلاك على طول الجانب المقابل للفجوة (الشكل 9.1) . التيار الكهرباني في السلك يولد مجالاً مغناطيسياً عبر الفجوة ، والذى بدوره يمغنط منطقة صغيرة على وَسط التسجيل ، وعكس أتجاه التيار يعكس أتجاه المغنطة على وَسط التسجيل .

آلية القراءة التقليدية تستغل حقيقة أن المجال المغناطيسي الذى يتحرك نسبة إلى **الملف يُنتج تياراً** كهربائياً في الملف . وعندما يمر سطح القرص تحت الرأس ، **فإنه يولد تياراً** بنفس القطبية التى سجل بها . وبُنية رأس القراءة في هذه الحالة **أساسا هى نفسها** فى الكتابة ، وبالتالي فإن الرأس نفسه يمكن إستخدامه فى الحالتين، **ويستخدم رأس واحد** في نظم القرص المرن والنظم القديمة للأقراص الصلبة .



الشكل (9.1) – رأس الكتابة بالحث والقراءة بالتحسس المغنطيسي

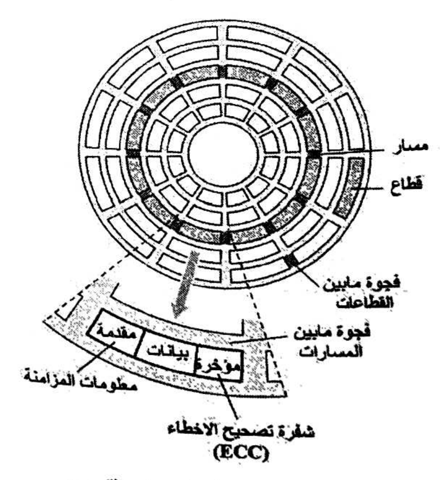
النظم الحديثة للأقراص الصلبة تستخدم آلية قراءة مختلفة ، حيث توجد رؤوس قراءة منفصلة مثبتة بالقرب من رأس الكتابة . رأس القراءة يتكون من حساس إستشعار مغناطيسي (MR – Magnetoresistive) محمي جزئيا ، ومادة الإستشعار (MR) لها مقاومة كهربائية تعتمد على إتجاه المغنطة فى الوسط المتحرك تحتها ، وبتمرير تيار خلال حساس الإستشعار يتم الكشف عن تغير المقاومة كإشارات جهد ، وتصميم حساس الإستشعار يسمح بالعمل على ترددات عالية مما يسمح بكثافة تخزين أكبر وسرعة تشغيل أعلى .

9.1.2 تنظيم البيانات وتنسيقها

الرأس هو جهاز صغير نسبيا قادر على القراءة من أو الكتابة على جزء من الطبق الدوار تحته ، وهذا يوضح أن تنظيم البيانات على الطبق عبارة عن مجموعة من

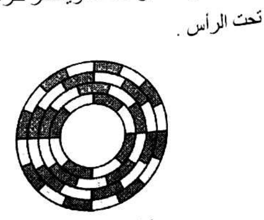
الحلقات الموحدة المركز وتدعى **المسارات** ، وكل مسار بنفس عرض الرأس **وتوجد الألاف** من المسارات على سطح الطبق ، ويصور الشكل (9.2) هذا **التخطيط للبيانات** . وتُفصل المسارات المتجاورة بواسطة **فجوات** (فجوة بينية) ، **وهذا يمنع أو على الأقل يقلل** من الاخطاء بسبب إختلال في الرأس أو تداخل **المجالات المغناطيسية** .

يتم نقل البيانات من وإلى القرص في **قطاعات** (الشكل 9.2) ، ويوجد مئات من **القطاعات على كل مسار** ، وهذه قد تكون إما ثابتة أو متغيرة الطول . وفي معظم **النظم المعاصرة** تستخدم قطاعات ذات طول ثابت بحجم (512) ثُمان تقريبا للقطاع **كقياس عالمي** وتُفصل القطاعات المتجاورة بفجوات .



الشكل (9.2) – توزيع البيانات على القرص

الخانات الثنائية القريبة من مركز القرص الدوار تنتقل من خلال نقطة ثابتة (مثل رأس القراءة/الكتابة) بسرعة أقل قليلا من الخانات التى على المحيط الخارجي ، ولذلك يجب أن توجد طريقة لتعويض هذا الأختلاف في السرعة بحيث يمكن للرأس قراءة جميع الخانات بنفس المعدل ، ويمكن أن يتم هذا عن طريق تثبيت عدد الخانات لكل المسارات فى القرص وزيادة التباعد مابين خانات البيانات المسجلة على مسارات القرص (كثافة الخانات متغيرة) ، ويمكن بعد ذلك فحص المعلومات بنفس المعدل عن طريق تدوير القرص بسرعة ثابتة تسمى **بالسرعة الزاوية الثابتة** (CAV-Constant Angular Velocity) حيث تقل كثافة الخانات كلما اتجهنا من المسارات الداخلية الى المسارات الخارجية وذلك للمحافظة على معدل نقل بيانات ثابت ، والشكل (9.3 – أ) يظهر تخطيط القرص بإستخدام السرعة الزاوية الثابتة ، وفيها يتم تقسيم القرص إلى عدد من القطاعات على شكل مثلث وإلى سلسلة من المسارات الموحدة المركز . ميزة إستخدام السرعة الزاوية الثابتة هو أنه يمكن عنونة قوالب فردية من البيانات بشكل مباشر بمعرفة المسار والقطاع لأي قالب . ولتحريك الرأس من موقعه الحالي إلى عنوان معين ، فإنه يأخذ حركة قصيرة من موضعه إلى المسار المحدد وينتظر فترة قصيرة مناسبة لكى يدور القطاع المطلوب تحت الرأس .



(ب) التسجيل المتعدد المناطق   (أ) السرعة الزاوية الثابتة

الشكل (9.3) – مقارنة لطرق تخطيط القرص

من عيوب السرعة الزاوية الثابتة هو أن كمية البيانات التي يمكن تخزينها على المسارات الطويلة الخارجية هو نفسه ما يمكن تخزينه على المسارات القصيرة الداخلية وتستعمل هذه الطريقة في القرص الصلب .

نتيجة أن الكثافة – ثنائي/خانة لكل مسافة خطية - تزداد مع الانتقال من المسار **الخارجي الأبعد الى المسار الداخلي الاقرب فإن سعة القرص التخزينية بنظام** **يطبق السرعة الزاوية الثابتة** مباشرة محدودة بالحد الاقصى لكثافة التسجيل التي **يمكن تحقيقها** على المسار الداخلي الأقرب . ولزيادة الكثافة فإن الأنظمة الحديثة **للقرص الثابت** تستخدام تقنية تعرف باسم **التسجيل المتعدد المناطق** ، والذي يقسم **السطح إلى** عدة مناطق متحدة المركز (نموذجيا 16) ، وداخل كل منطقة عدد **الثنائيات لكل** مسار ثابت ، فالمناطق البعيدة عن المركز تحتوى على خانات أكثر **(أكثر قطاعات)** من المناطق القريبة إلى المركز ، وهذا يسمح بمزيد من السعة **التخزينية الإجمالية** على حساب - إلى حد ما - تعقيد أكثر في الدوائر الإلكترونية . **فكلما تحرك** الرأس من منطقة إلى أخرى طول الخانة يتغير (على طول المسار) **مما يسبب تغيير** في زمن القراءة والكتابة . والشكل (9.3 – ب) يشير لطبيعة **التسجيل المتعدد المناطق ؛ وفي هذا التوضيح فإن كل منطقة** بعرض مسار واحد **فقط ، ونلاحظ هنا أن السرعة الزاوية للقرص تزداد كلما تحرك** الرأس من **المسارات الخارجية الى المسارات الداخلية** للمحافظة على نفس معدل نقل البيانات **من تحت الرأس وتسمى تقنية هذه الطريقة في التسجيل بالسرعة الخطية الثابتة (CLV-Constant Linear Velocity) وتستعمل هذه التقنية في الأقراص الضوئية .**

في القرص - بديهياً - يجب أن تكون هناك نقطة بداية **على المسار و وسيلة لتحديد** بداية ونهاية كل قطاع ، ويتم تحقيق هذه المتطلبات **عن طريق تسجيل بيانات التحكم** على القرص ، وبالتالي يتم تنسيق القرص **مع معلومات إضافية تستخدم فقط من** قبل مشغل القرص ولا يمكن للمستخدم من الوصول إليها **وذلك للتحكم وتنسيق** القرص .

9.1.3 الخصائص المادية للأقراص المغناطيسية

يسرد الجدول (9.1) الخصائص الرئيسية التي تميز مابين **الأنواع المختلفة من** الأقراص المغناطيسية . أولا ، قد يكون الرأس إما ثابتاً أو متحركاً **بالنسبة الى** الاتجاه الشعاعي للطبق . في **القرص الثابت الرأس** هناك رأس واحد **للقراءة** والكتابة لكل مسار ، وكل الرؤوس مثبتة على ذراع ثابتة **تمتد على جميع المسارات** وهذه النظم نادرة هذه الايام . في **القرص المتحرك الرأس** يوجد رأس **واحد فقط** للقراءة والكتابة ، ومرة أخرى هذه الرأس مثبتة على **ذراع ولكن لأن الرأس يجب** أن تكون له القدرة على التموضع فوق أي مسار **فأن الذراع يمكن أن تتمدد أو** تتراجع لهذا الغرض ، وكما في الشكل (9.4) .

القرص نفسه مثبت على محرك الأقراص والذي يتألف من **ذراع ومغزل (محرك)** يدوِّر القرص ، والالكترونيات اللازمة للمدخلات والمخرجات **من البيانات الثنائية.**

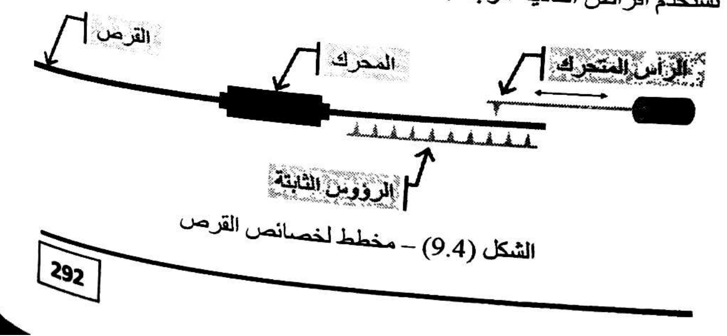
القرص الثابت (غير القابل للفك/النزع) **هو المثبت بشكل دائم على محرك الأقراص،** والقرص الصلب في الحاسوب الشخصي **هو مثال على ذلك حيث أنه** قرص غير قابل للفك . **القرص المتحرك** (القابل للفك/النزع) **يمكن إزالته وإستبداله** بقرص آخر ، **ومن ميزاته أن عدداً محدوداً من الأنظمة القرصية يمكن أن يوفر**

كميات غير محدودة من البيانات ، وعلاوة على ذلك فقد يتم نقل هذا القرص من **نظام حاسب إلى آخر** ، والأقراص المرنة هي مثال على الأقراص المتحركة .

الجدول (9.1) – الخصائص المادية للأنظمة القرصية وأنواعها

الأطباق :	**حركة الراس:**
بطبق فردى	برأس ثابت (واحد لكل مسار)
متعدد الأطباق	برأس متحرك (واحد لكل سطح)
آلية الرأس:	**قابلية القرص للحمل:**
متصل (المرن)	قرص ثابت
فجوة ثابتة	قرص متحرك
فجوة هوائية (القرص الصلب)	
الأوجه:	
فردى الوجه	
مزدوج الوجه	

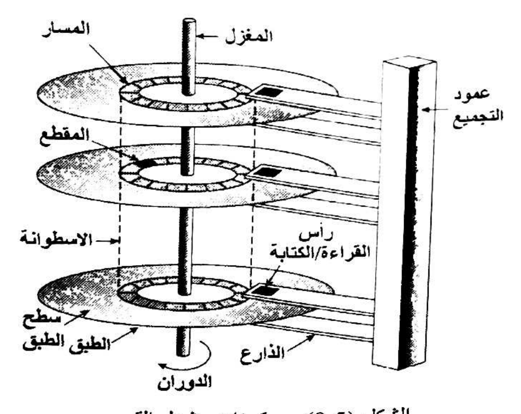
بالنسبة لمعظم الأقراص يطلى جانبى الطبق بطلاء ممغنط لكى تصبح مزدوجة الوجه (يتم التخزين على سطحى القرص) ، وبعض الأنظمة القرصية الأقل تكلفة تستخدم أقراص احادية الوجه .



الشكل (9.4) – مخطط لخصائص القرص

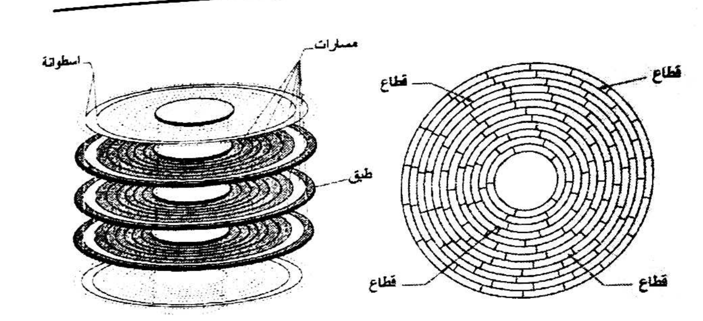
بعض محركات الأقراص تستوعب **أطباق متعددة** مكدسة عموديا فوق بعضها وتبعد عن بعضها بجزء من البوصة مع وجود اذرع متعددة فوقها (أنظر الشكل – 9.5).

والأقراص متعددة الأطباق توظف رؤوس متحركة ، ومع رأس واحد للقراءة والكتابة على سطح كل طبق . وجميع الرؤوس مثبتة ميكانيكياً بحيث تكون كلها على مسافة واحدة من مركز القرص وتتحرك معا ، ولهذا في أي وقت كل الرؤوس تتموضع على مسارات متساوية من مركز القرص (أسطوانة) .



الشكل (9.5) – مكونات مشغل القرص

ويشار إلى مجموعة المسارات التى في نفس الموضع النسبى على الأطباق على أنها **أسطوانة** ، وعلى سبيل المثال كل من المسارات المظللة في الشكل (9.6) هي جزء من اسطوانة واحدة .



الشكل (9.6) – المسار والمقطع والأسطوانة والطبق فى نظام القرص

وأخيرا ، فإن آلية الرأس تصنف الأقراص إلى ثلاثة أنواع . الأول ، رأس القراءة والكتابة موضوع على مسافة ثابتة فوق الطبق مما يسمح بفجوة ثابتة من الهواء . الثاني ، الرأس يتصل ماديا مباشرة مع الوسط (سطح الطبق) خلال عملية القراءة أو الكتابة ويتم إستخدام هذه الآلية مع **القرص المرن** حيث إنه صغيرة وذو طبق مرن وأقل أنواع الأقراص تكلفة .

ولفهم النوع الثالث من الاقراص فنحن بحاجة إلى التعليق على العلاقة بين كثافة **البيانات وحجم فجوة الهواء** . الرأس يجب أن يولد أو يتحسس المجال **الكهرومغناطيسي** بحجم كاف حتى تتم الكتابة والقراءة بشكل صحيح . وعندما **يكون الرأس صغيراً** يستوجب ذلك أن يكون أقرب ما يكون إلى سطح الطبق حتى **يعمل على نحو صحيح** ، ورأس صغير يعنى أيضا مسارات ضيقة مما يزيد من **كثافة البيانات وهذا أمر مرغوب فيه** ، ولكن إن قرب الرأس من القرص يزيد من **احتمال الخطأ نتيجة الشوائب أو العيوب** . ومع تقدم التقنية تم تصنيع قرص **"وينشستر" (القرص الصلب الحديث)** بحيث أن الرؤوس فى قرص **وينشستر**

مجمعة على محرك أقراص مغلق وفي محيط شبه خالي من الشوائب ، وهي مصممة لتعمل أقرب إلى سطح القرص من رؤوس الاقراص الصلبة التقليدية وبالتالي تسمح بكثافة أكبر للبيانات . والرأس هو في الواقع عبارة عن رقيقة معدنية هوائية تستقر برفق على سطح الطبق . وعندما يتحرك سطح القرص فإن ضغط الهواء المتولد بواسطة دوران القرص (المغزل) يكفي لأن يجعل الرأس مستقراً أو عائماً فوق سطح الطبق .

9.1.4 عوامل أداء القرص المغناطيسى

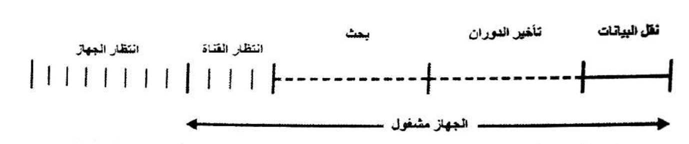
يظهر الرسم التخطيطي في الشكل (9.7) الأزمنة اللازمة لأنتقال البيانات فى القرص فى حالة الإدخال/الإخراج . فعندما يشتغل محرك الأقراص يدور القرص بسرعة ثابتة ، وللقراءة أو الكتابة يجب أن يوضع الرأس على المسار المطلوب وعلى بداية القطاع المطلوب فى المسار .

أن أختيار المسار يتضمن تحريك الرأس في الأنظمة المتحركة الرأس أو أختيار الرأس فى الانظمة ذات الرؤوس الثابتة . في الانظمة المتحركة الرأس فإن الزمن المستغرق لوضع الرأس على المسار المطلوب يدعى **زمن البحث** . وفي كلتا الحالتين (متحرك أو ثابت) فإنه عندما يتم تحديد المسار تنتظر وحدة تحكم القرص حتى يدور القطاع المناسب ليصطف تحت الرأس .

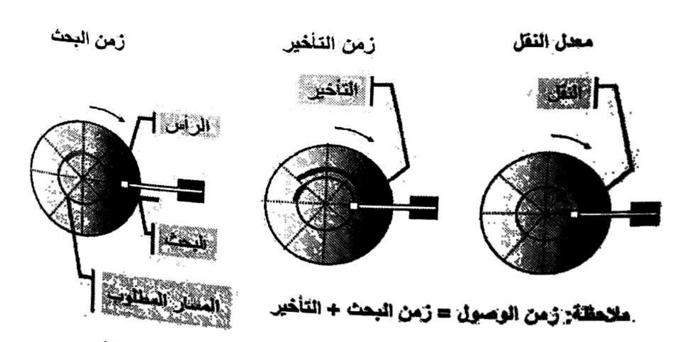
الزمن المستغرق حتى تصل بداية القطاع المستهدف إلى تحت الرأس يدعى **زمن تأخير الدوران** . ومجموع زمن البحث - إن وجد - مع تأخير الدوران يساوي **زمن الوصول** ، وهو الزمن اللازم للوصول الى موضع القراءة أو الكتابة .

عندما يصل الرأس الى الموضع المطلوب يتم تنفيذ عملية القراءة أو الكتابة حيث يتحرك القطاع تحت الرأس ، وفى هذا الجزء يتم نقل البيانات فى العملية ، والزمن

اللازم للنقل هو زمن النقل والشكل (9.8) يوضح ذلك . بالإضافة إلى زمن **الوصول وزمن النقل** هناك عدة عوامل تأخير ترتبط عادة مع عمليات **الإدخال/الإخراج** القرصية . فعندما تصدر عملية ما طلب إدخال/إخراج يجب أولاً **أن تنتظر في** طابور حتي يكون الجهاز متاحاً لها ، وعندها يتم تخصيص الجهاز **لهذه العملية ، وإذا كان** الجهاز مشتركاً فى قناة إدخال/إخراج واحدة أو مجموعة **قنوات إدخال/إخراج** مع مشغلات أقراص أخرى فعندها ربما يكون هناك إنتظار **إضافي لكى تتوفر** القناة ، وبعدها يتم تنفيذ البحث لبدء التواصل مع القرص ، وكما **هو مبين بالمخطط** الزمنى بالشكل (9.7).



الشكل (9.7) – أزمنة أنتقال بيانات الإدخال/الإخراج فى القرص



الشكل (9.8) – العوامل الأساسية لقياس أداء القرص المغناطيسى

زمن البحث هو الزمن المطلوب لتحريك ذراع القرص للمسار المطلوب . وزمن البحث يتكون من جزئيين رئيسيين : زمن بدء التشغيل ، والزمن المستغرق لإجتياز المسارات التي يجب أن تُجتاز عندما يكون الذراع فى السرعة المناسبة ، وللأسف فإن زمن الإجتياز ليس خطيا مع عدد المسارات ، ولكنه يشمل زمن التسوية (الزمن بعد وضع الرأس على المسار المستهدف وحتى يتم التأكد من مُعرف المسار) ، والمتوسط النموذجي لزمن البحث على الأقراص الصلبة هو أقل من 10 ملى ثانية.

تأخير الدوران للأقراص : فى غير الأقراص المرنة تدور الأقراص بسرعات تتراوح من 3600 دورة في الدقيقة الى ما يصل 20000 دورة في الدقيقة ، وفي السرعة الأخيرة هناك دورة واحدة لكل 3 ملى ثانية ، وفي المتوسط يكون زمن تأخير الدوران بقيمة 1.5 ملى ثانية.

زمن النقل : زمن النقل من وإلى القرص يعتمد على سرعة الدوران وهى كالتالي:

$$T = \frac{b}{rN}$$

حيث T – زمن النقل ، b – عدد الثُمان التى ستُنقل ، N – عدد الثُمان على المسار ، r – سرعة الدوران (دورة لكل ثانية) . وبذلك فإن إجمالى متوسط زمن الوصول يكون كالتالى :

$$T_a = T_s + \frac{1}{2r} + \frac{b}{rN}$$

حيث T_s – **هى متوسط زمن البحث ، ولاحظ أن فى مشغل الأقراص المقسم لعدة مناطق عدد الثُمان لكل مسار متغير وهذا يعقد العملية الحسابية .**

الجدول (9.2) يعرض المواصفات النموذجية لعدة أنواع من الأنظمة القرصية ومعدلات الأداء وإستخدامات كل نوع .

الجدول (9.2) – المواصفات النموذجية للأقراص المعاصرة العالية الأداء .

نوع القرص	Seagate ES.2	Seagate 7200.10	Seagate 7200.9	Seagate	Hitachi Microdrive
الأستخدام	خوادم ذات السعة الكبيرة	الحواسيب عالية الأداء	الحواسيب العادية	الحواسيب المحمولة	الاجهزة اليدوية
السعة	1 تيرا خانة	750 غيغا خانة	160 غيغا خانة	120 غيغا خانة	8 غيغا خانة
أدنى زمن بحث من مسار الى مسار	0.8 ملى ثانية	0.3 ملى ثانية	1.0 ملى ثانية	-	1.0 ملى ثانية
متوسط زمن البحث	8.5 ملى ثانية	3.6 ملى ثانية	9.5 ملى ثانية	12.5 ملى ثانية	12 ملى ثانية
سرعة المحرك	7200 لفة فى الثانية	7200 لفة فى الثانية	7200 لفة فى الثانية	5400 لفة فى الثانية	3600 لفة فى الثانية
متوسط تأخير الدوران	4.16 ملى ثانية	4.16 ملى ثانية	4.17 ملى ثانية	6.5 ملى ثانية	8.33 ملى ثانية
اعلى معدل النقل	3 غيغا ثمان/للثانية	300 ميغا ثمان/للثانية	300 ميغا ثمان/للثانية	150 ميغا ثمان/للثانية	10 ميغا ثمان/للثانية
عدد الخانات لكل قطاع	512	512	512	512	512
عدد المسارات لكل اسطوانة (عدد الأسطح)	8	8	2	8	2

9.2 الصفوف المتكررة من الأقراص المستقلة (RAID)

إن معدل التحسين في أداء وحدات التخزين الثانوي اقل منه بكثير عن معدل تحسين الأداء في المعالجات والذاكرة الرئيسية ، وربما عدم التطابق هذا جعل نظام التخزين القرصي المحور الرئيسي في مساعى تحسين وتطوير الاداء العام لنظام الحاسب .

إن التقدم التقني في التخزين القرصي ادي الى تطوير صفوف من الأقراص المتعددة التي تعمل بشكل مستقل ومتواز . وفي هذه الأقراص المتعددة ، طالما أن البيانات المطلوبة موجودة على اقراص منفصلة هذا يُمكن من معالجة طلبات إدخال/إخراج متعددة بالتوازي ، وعلاوة على ذلك فإن عملية إدخال/اخراج واحدة يمكن تنفيذها بالتوازي حتى ولو كانت قوالب البيانات المراد الوصول إليها موزعة على عدة أقراص .

بإستخدام الأقراص المتعددة ، هناك مجموعة متنوعة و واسعة من الطرق التي يمكن من خلالها تنظيم البيانات وتكرارها لتحسين الأعتمادية ، وهذا صعّب من إمكانية وضع خطط لقاعدة بيانات واحدة بحيث يمكن إستخدامها في عدة منصات وأنظمة تشغيل . ومع التقدم العلمي في مجال الحاسبات وُفقت الأبحاث الصناعية لنظام موحد لتصميم قاعدة بيانات تستخدم الأقراص المتعددة فى الذاكرة الثانوية لأنظمة الحاسب الحديثة ، ويُعرف باسم **الصفوف المتكررة من الأقراص المستقلة (RAID - Redundant Array of Independent Disks)** . ونظام الصفوف المتكررة من الأقراص المستقلة (RAID) يتكون من سبعة مستويات ، من صفر الى ستة ، وهذه المستويات لا تعني وجود علاقة هرمية فيما بينها ولكن تُوضح بُنية تصاميم مختلفة تتقاسم ثلاث خصائص مشتركة :-

1. **نظام الصفوف المتكررة من الأقراص المستقلة (RAID) هو طقم من محركات الأقراص الفعلية** (مادياً أقراص صلبة عادية) ينظر إليها عملياً (**إفتراضيا**) من قبل نظام التشغيل كمحرك (مشغل) أقراص واحد.
2. **يتم توزيع البيانات على طقم من محركات الأقراص الفعلية التي في صف واحد على شكل خطي تسمى أجزاء** ، ومجموعة الأجزاء المتتالية التي **على المحركات المتجاورة في الصف تسمى شريط** (عدد الأجزاء بالشريط **هو عدد الأقراص في الصف**).
3. تستخدم السعة الزائدة للأقراص في تخزين معلومات التصحيح والتحكم **في الأخطاء** (خانات التماثل - Parity bits) التي تضمن استرداد البيانات **في حالة حدوث قصور أو تلف لقرص ما**.

تفاصيل الخاصية الثانية والثالثة تختلف بإختلاف مستويات نظام الصفوف **المتكررة من الأقراص المستقلة** ، فالمستوى - 0 (RAID-0) والمستوى - 1 **(RAID-1) لا تدعم الخاصية الثالثة**.

إستراتيجية الصفوف المتكررة من الأقراص المستقلة (RAID) توظف مشغلات **أقراص متعددة وتوزع البيانات** بطريقة تُمكن من الوصول إلى البيانات في وقت **واحد من مشغلات أقراص متعددة** ، وبالتالي تحسن أداء الإدخال/الإخراج وتسمح **بسهولة إضافة الزيادات**.

المساهمة الفريدة لتقنية الصفوف المتكررة من الأقراص المستقلة هي الإستخدام الفعال لميزة التكرار ، وكذلك السماح للرؤوس والمشغلات المتعددة بالعمل في وقت واحد مما يحقق معدل إدخال/إخراج ونقل عاليين ، ولكن إستخدام أجهزة متعددة يزيد من إحتمال الأخفاق ، وللتعويض عن هذا الأنخفاض في الدقة فإن

الصفوف المتكررة من الأقراص المستقلة تخزن معلومات تُستخدم لتصحيح الأخطاء والتي تُمكن من استرداد البيانات المفقودة بسبب إخفاق قرص ما.

الجدول (9.3) يقدم ملخصاً لمستويات نظام الصفوف المتكررة من الأقراص المستقلة (RAID) السبعة. فالجدول يظهر كفاءة الإدخال/الإخراج سواء من حيث سعة نقل البيانات ، القدرة على نقل البيانات ، ومعدل طلب الإدخال/الإخراج أو القدرة على تلبية طلبات الإدخال/الإخراج ، ويتم تمييز قوة كل مستوى من الصفوف المتكررة من الأقراص المستقلة بالتظليل المناسب في الجدول.

في الفقرات التالية سنستعرض المستويات السبعة لنظام الصفوف المتكررة من الأقراص المستقلة ببعض التفصيل وستبين الأشكال الموضحة لهذه المستويات مثلاً لنظام الصفوف المتكررة من الأقراص المستقلة يدعم حجم بيانات مستخدم (بيانات غير مكررة) يتطلب أربعة أقراص فعلية ، والأشكال ستسلط الضوء على توزيع بيانات المستخدم والبيانات المضافة وتشير إلى متطلبات التخزين النسبية لكل مستوى.

9.2.1 المستوى-0

المستوى-0 من تقنية الصفوف المتكررة من الأقراص المستقلة **(RAID) لا يتضمن تكراراً** (تكرار بيانات المستخدم) **وذلك لتحسين أداء بعض التطبيقات القليلة كالموجودة مثلاً على أجهزة الحاسب العملاقة حيث الأداء والسعة من الأهتمامات الأولية ، وكذلك لأن التكلفة المنخفضة أكثر أهمية من تحسين الأعتمادية**.

الجدول (9.3) - دليل المستويات السبعة لنظام الصفوف المتكررة من الأقراص المستقلة (RAID)

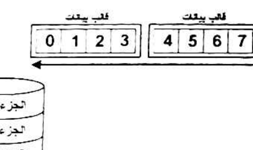
الصنف	المستوى	الوصف	الأقراص المطلوبة	مقدار توفر البيانات	سعة نقل بيانات الادخال/الاخراج الكبيرة	معدل طلب للادخال/الاخراج المحدود
تجزئة	0	بدون تكرار	ن	أقل من قرص فردى	عالى جداً	عالى جداً فى القراءة و الكتابة
مرايا	1	مرآة	2 × ن	أعلى من 2 (RAID) او 3 أو 4 أو 5 ، واقل من 6 (RAID)	أعلى من قرص فردى فى القراءة، ومساوى لقرص فردى فى الكتابة	الضعف فى القراءة نسبة للقرص الفردى ، ومساوى للقرص الفردى فى الكتابة
الوصول المتوازى	2	تكرار باستخدام شفرة - هامنك	ن + س	أعلى بكثير من قرص فردى ، قريب من 3 (RAID) أو 4 أو 5	الأعلى فى كل الخيارات المذكورة	ضعف القرص الفردى تقريبا
	3	تماثل باستخدام الخانات المتداخلة	ن + 1	أعلى بكثير من قرص فردى ، قريب من 2 (RAID) أو 4 أو 5	الأعلى فى كل الخيارات المذكورة	ضعف القرص الفردى تقريبا
الوصول المستقل	4	تماثل باستخدام القالب المتداخل	ن + 1	أعلى بكثير من قرص فردى ، قريب من 2 (RAID) أو 3 أو 5	مساوى (RAID) 0 فى القراءة ، أقل بكثير من قرص فردى فى الكتابة	مساوى (RAID) 0 فى القراءة ، أقل بكثير من قرص فردى فى الكتابة
	5	تماثل موزع بالقالب المتداخل	ن + 1	أعلى بكثير من قرص فردى ، قريب من 2 (RAID) أو 3 أو 4	مساوى (RAID) 0 فى القراءة ، اقل بكثير من قرص فردى فى الكتابة	مساوى (RAID) 0 فى القراءة ، أقل من قرص فردى فى الكتابة
	6	تماثل مزدوج التوزيع بالقالب المتداخل	ن + 2	الأعلى فى كل الخيارات المذكورة	مساوى (RAID) 0 فى القراءة ، أقل من (RAID) 5 فى الكتابة	مساوى (RAID) 0 فى القراءة ، أقل بكثير من (RAID) 5 فى الكتابة

ن = عدد أقراص البيانات ، س تتناسب طرديا مع لوغاريتم ن .

فى المستوى-0 لتقنية الصفوف المتكررة من الأقراص المستقلة يتم توزيع بيانات المستخدم والنظام على كافة الأقراص في الصفوف ، وهذا له ميزة بارزة عن إستخدام قرص واحد كبير : لو كان هناك انتظار لطلبي إدخال/إخراج مختلفين فى قوالب مختلفة من البيانات هناك فرصة جيدة بأن القوالب المطلوبة موجودة على أقراص مختلفة ، وهكذا فإن الطلبين يمكن أن يرسلا بالتوازي مما يحد من زمن الإنتظار .

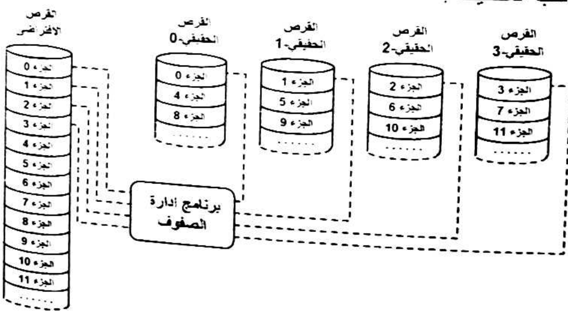
فى المستوى-0 لا يتم توزيع البيانات ببساطة على صف من الأقراص : البيانات تُجزأ (أجزاء) على الأقراص المتوفرة ، ويمكن فهمها أفضل من خلال النظر الى الشكل (9.9) ، وينظر الى بيانات المستخدم والنظام كافة كما لو أنها خزنت على قرص واحد (إفتراضياً) . ويتم تقسيم القرص الإفتراضى إلى أجزاء ، وهذه الأجزاء قد تكون قوالب مادية ، أو قطاعات ، أو وحدة أخرى . وهذه الأجزاء مسقطة بشكل متسلسل على أقراص فعلية متتالية في صفوف من نظام الصفوف المتكررة من الأقراص المستقلة ، ويشار إلى مجموعة من الأجزاء الإفتراضية المتتالية المسقطة كأجزاء على صف بانها **شريط** .

ففي صف مكون من عدد ن – قرص ، أول مجموعة (بعدد س جزء) من الأجزاء الإفتراضية تخزن فعليا كأول جزء على كل قرص (من مجموعة ن قرص) ، وتشكل شريط الخط الأول ؛ ثاني مجموعة من الأجزاء توزع كشريط الخط ثاني على كل قرص وهكذا ، وميزة هذا التصميم أنه لو كان هناك طلب إدخال/إخراج واحد يتضمن عدة أجزاء إفتراضية متجاورة يمكن التعامل مع الشريط - ن بالتوازى ، وهذا يحد كثيراً من زمن النقل فى الإدخال/الإخراج .



الشكل (9.9) - المستوى–0 من نظام الصفوف المتكررة من الأقراص المستقلة (RAID-0) – بدون تكرار

الشكل (9.10) يشير إلى إستخدام برامج لإدارة الصفوف فى مطابقة مساحة القرص مابين الإفتراضية والفعلية للمستوى – 0 من نظام الصفوف المتكررة من الأقراص المستقلة ، وهذا البرنامج ينفذ إما في النظام الفرعي للقرص أو في الحاسب المضيف .



الشكل (9.10) - إستخدام برامج لإدارة الصفوف فى مطابقة مساحة القرص مابين الأفتراضية والفعلية للمستوى – 0

9.2.2 المستوى-1

فى المستويات الأخرى لتقنية الصفوف المتكررة من الأقراص المستقلة يتم إستخدام شكل من أشكال تصحيح الأخطاء (حسب التمثل) في التكرار (بيانات مضافة – زيادة) ، ففي المستوى-1 من الصفوف المتكررة من الأقراص المستقلة يتحقق التكرار بواسطة مضاعفة (تكرار) كافة البيانات ، وكما هو موضح فى الشكل (9.11) .

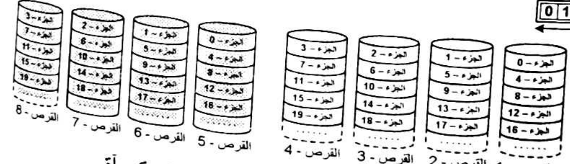
في المستوى-1 يتم إستخدام أسلوب تشريط البيانات ، وكما هو الحال في في المستوى-0 ، ولكن في هذه الحالة كل جزء إفتراضي يتم مطابقته على قرصين فعليين منفصلين بحيث أن كل قرص في الصفوف له قرص آخر مرآة له ويحتوي على نفس البيانات ، والمستوى-1 يمكن انجازه بدون تشريط البيانات لكن هذا غير شائع . وهناك عدد من الجوانب الإيجابية للمستوى-1 من نظام الصفوف المتكررة من الأقراص المستقلة ، ومنها :

1. إمكانية تقديم الخدمة لطلب قراءة من قبل أحد القرصين المحتويين على البيانات المطلوبة ، وذلك بناء على أيهما يوفر الحد الأدنى من زمن البحث وزمن التأخير .
2. طلب الكتابة يتم بتحديث الجزئين المتناظرين ولكن هذا ينبغي القيام به في نفس الوقت ، وبالتالى فإن أداء الكتابة يُحَدد من قبل أبطأ كتابة فى عمليتى الكتابة (أي تلك التي تنطوي على أكبر زمن بحث و زمن تأخير) ، ومع ذلك لا يوجد "ضريبة كتابة" فى المستوى-1 . المستويات من 2 الى 6 من الصفوف المتكررة من الأقراص المستقلة تنطوي على إستخدام خانة

التماثل (Parity bit) ولذلك فعندما يتم تحديث شريط يجب حساب خانة التماثل ومن ثم تحديثها .

3. سهولة التعافي من الفشل ، فعند إخفاق قرص ما يمكن الوصول إلى البيانات من القرص الثاني (المرآة له / المناظر) .

العيب الرئيسي للمستوى-1 من نظام الصفوف المتكررة من الأقراص المستقلة هي **التكلفة** ، حيث يتطلب مساحة قرص ضعف مساحة القرص الإفتراضي الذي يدعمه وبسبب ذلك فإن إستخدامه مقصوراً على الأقراص التي تقوم بتخزين بيانات وبرامج النظام وغيرها من الملفات الهامة ، وفي هذه الحالات المستوى-1 من نظام **الصفوف المتكررة** من الأقراص المستقلة يوفر من الزمن الحقيقي اللازم لنسخ **جميع البيانات** حيث إنه في حالة حدوث فشل للقرص كل البيانات الهامة متاحة فوراً على الأقراص المرآة/المناظر .



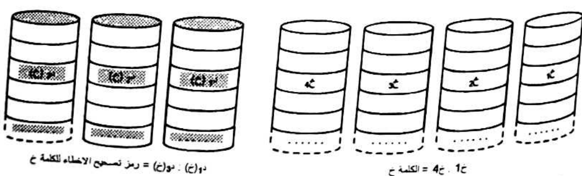
الشكل (9.11) - المستوى-1 (RAID-1) – تكرار كمرآة

9.2.3 المستوى-2

المستويان 2 و 3 من نظام الصفوف المتكررة من الأقراص المستقلة يستخدمان تقنية الوصول المتوازي . وفي الصفوف ذات الوصول المتوازى فإن كل الأقراص تساهم في تنفيذ طلب الإدخال/الإخراج ، وعادةً ماتتم مزامنة مغزل كل مشغلات

الأقراص بحيث في أي وقت من الأوقات يكون رأس كل قرص فى نفس الموضع لكل الأقراص . وكما هو الحال في المخططات الأخرى لنظام الصفوف المتكررة من الأقراص المستقلة فإنه يتم إستخدام أسلوب تشريط البيانات .

في حالة المستوى-2 والمستوى-3 من نظام الصفوف المتكررة من الأقراص المستقلة الأجزاء صغيرة جدا ، وغالبا بصغر 8 – خانات (ثُمان) أو كلمة واحدة . وفى المستوى-2 يحسب رمز تصحيح الأخطاء لخانات البيانات (الجزء) فى كل قرص من الاقراص ، ويتم تخزين الرمز (الخاص بالشريط/الأجزاء) فى المواضع المخصصة لذلك على أقراص متعددة خاصة بحفظ رمز تصحيح الأخطاء ، وعادة ما يتم إستخدام رمز "هامنك" فى حساب رمز تصحيح الأخطاء وهو قادر على كشف خطأ خانتين وتصحيح خطأ خانة واحدة ، وكما هو موضح في الشكل (9.12) . وبالرغم من أن المستوى-2 يتطلب عدد أقراص أقل من المستوى-1 لكنه لا يزال مكلفا جداً ، والمستوى-2 يكون خياراً فعالاً فقط في البيئة التي تحدث بها أخطاء قرص كثيرة . وحاليا ، نظراً للموثوقية العالية للأقراص ومشغلات الأقراص فإن نظام المستوى-2 يعتبر مبالغاً فيه ولا يُطبق .



القرص - 1 القرص - 2 القرص - 3 القرص - 4 القرص - 5 القرص - 6 القرص - 7

الشكل (9.12) - المستوى–2 (RAID-2) – تكرار بحفظ شفرة التحكم بالأخطاء (شفرة هامنك)

9.2.4 المستوى-3

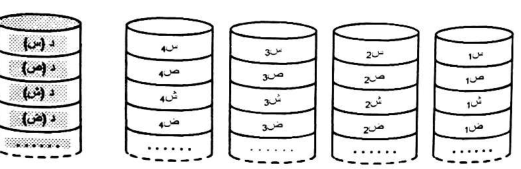
تنظيم المستوى-3 يماثل المستوى-2 ، والفرق هو أن المستوى-3 يتطلب إضافة قرص واحد فقط (تكرر قرص) بغض النظر عن كبر مصفوفة الأقراص . **المستوى-3** يستخدم الوصول المتوازي للبيانات الموزعة على أجزاء صغيرة ، وبدلاً من رمز تصحيح الأخطأ يتم حساب خانة تماثل بسيطة للخانات الموجودة في نفس الموقع على جميع أقراص البيانات وحفظها في الخانة المناظرة لها في قرص التماثل (المضاف) بحيث ينتج عن ذلك حفظ خانات التماثل للشريط الناتجة من خانات الأجزاء المتناظرة على كل الاقراص ، وخانة التماثل الواحدة تنجت من حساب يضم خانات البيانات الموجودة في نفس الموقع ولكن على أجزاء مختلفة من نفس الشريط (كل جزء خانة) ، وكما هو موضح في الشكل (9.13) .

التكرار : في حالة فشل مشغل القرص فإنه يتم التواصل مع مشغل قرص التماثل، ويعاد بناء البيانات من مشغلات الأقراص المتبقية بإستخدام خانات التماثل للشريط وخانات البيانات المتناظرة للأجزاء في الأقراص العاملة (المتبقية) . وعندما يتم إستبدال مشغل القرص الفاشل يعاد تخزين البيانات المفقودة على مشغل القرص الجديد وتستأنف العملية .

في حالة فشل قرص لا تزال كافة البيانات متاحة بشكل يسمى بالصيغة المصغرة . وفي هذا الوضع ، للقراءة فيتم إعادة إنشاء البيانات المفقودة على السريع بإستخدام حساب يستغل خانة التماثل وبقية خانات البيانات المتبقية لأعادة إنتاج الخانة المفقودة . وعندما تتم الكتابة في هذا الوضع يجب الحفاظ على تطابق خانة التماثل الجديدة (المنتجة من الأقراص العاملة) مع السابقة حتى يمكن إنشاء بيانات كاملة في وقت لاحق .

للعودة إلى صيغة العمل الطبيعي يتطلب ذلك إستبدال القرص الفاشل ويتم إعادة إنشاء كامل محتوياته على القرص الجديد .

الأداء : لأن البيانات مُجزئة في أجزاء صغيرة جداً ، فالمستوى-3 يمكنه تحقيق معدلات نقل بيانات عالي جداً ، وأي طلب إدخال/إخراج ينطوي على نقل متوازي للبيانات من كافة اقراص البيانات ، ويلاحظ التحسن في الأداء خاصة عند نقل كميات كبيرة من البيانات ، ولكن من جهة أخري فإن تنفيذ طلب إدخال/إخراج واحد فقط يَحد من الأداء .

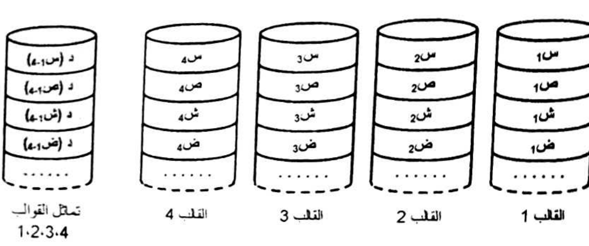


الشكل (9.13) - المستوى–3 (RAID-3) – تكرار بحفظ شفرة التحكم بالأخطاء (خانات تماثل متداخلة - للاجزاء بالكامل)

9.2.5 المستوى-4

في مستويات من 4 إلى 6 تمت الإستفادة من تقنية الوصول المستقل . **وفي الصفوف المستقلة الوصول كل قرص عضو يعمل بشكل مستقل بحيث يمكن لطلبات** إدخال/إخراج منفصلة ان تُؤدى بالتوازى . للسبب السالف الذكر **فإن الصفوف المستقلة الوصول هي أكثر ملائمة للتطبيقات التي لها معدل طلبات إدخال/إخراج مرتفعة ونسبيا أقل ملائمة للتطبيقات التي تتطلب معدلات نقل بيانات عالية .**

كما هو الحال في باقى أنظمة الصفوف المتكررة من الأقراص المستقلة يتم إستخدام أسلوب تشريط البيانات . وفي حالة المستوى-4 إلى المستوى-6 الأجزاء كبيرة **نسبيا** . **فى المستوى-4** يتم حساب شريط التماثل خانة بخانة من خلال شريط **البيانات** (الأجزاء) المناظر لها فى كل اقراص البيانات ، وتخزن خانات التماثل **فى الشريط** المناظر على قرص التماثل ، وكما هو موضح في الشكل (9.14) . **المستوى-4** ينطوي على ضريبة كتابة عندما يتم تنفيذ طلب كتابة إدخال/إخراج **صغير الحجم** ، ففي كل مرة تحدث كتابة يجب تحديث البيانات وكذلك خانات **التماثل** المناظرة (الشريط) .



الشكل (9.14) - المستوى-4 (RAID-4) – تكرار بحفظ شفرة التحكم بالأخطاء (خانات تماثل متداخلة – بمستوى قالب)

9.2.6 المستوى-5

المستوى-5 منظم بشكل مشابه إلى المستوى-4 ، والفرق هو أن المستوى-5 يوزع **شرائط التماثل على كافة الأقراص** ، والطريقة النموذجية لذلك هو اسلوب **حلقة روبين** ، وكما هو موضح في الشكل (9.15) . فلعدد محدد من الأقراص فإن شريط **التماثل - ن يوضع على قرص مختلف عن الذى يحتوى أجزاء الشريط - ن** ويتم

تكرار النمط بعد ذلك . إن توزيع شرائط التماثل على كافة المشغلات (الأقراص) **يجنب إحتمال** خطر عنق الزجاجة بالنسبة للإدخال/الإخراج والموجود في **المستوى-4** .

القرص - 1 القرص - 2 القرص - 3 القرص - 4 القرص - 5

الشكل (9.15) - المستوى-5 (RAID-5) – تكرار بحفظ شفرة التحكم بالأخطاء (خانات تماثل متداخلة – بمستوى قالب – موزع)

9.2.7 المستوى-6

في نظام المستوى-6 فإن حساب التماثل يتم بطريقتين مختلفتين وتخزن في قوالب **منفصلة على أقراص** مختلفة ، وهكذا فإن نسق نظام المستوى-6 الذى يتطلب عدد **ن** - قرص لبيانات مستخدم يتكون فعلياً من ن+2 - قرص ، والشكل (9.16) **يوضح ذلك** .

ونوجد خوارزميتان مختلفتين لفحص بيانات ، وإحدهما هى حساب منطق بوابة **أو-الحصرية** (XOR) المستخدمة في المستوى-4 والمستوى-5 ، والأخرى هى **خوارزمية مستقلة** لفحص البيانات ، وهذا يُمكن من إعادة إنشاء البيانات حتى إذا **فشل قرصين يحتويان على بيانات المستخدم** .

من ميزات المستوى-6 هو أنه يقدم إمكانية عالية جداً لتوفر البيانات ، فثلاثة أقراص يجب أن تفشل في غضون فترة متوسط زمن الإصلاح (MTTR) لكى يتسبب ذلك فى فقدان البيانات . ومن جهة أخرى يتحمل نظام المستوى-6 ضريبة **كتابة القوالب الكبيرة لأن كل كتابة تؤثر فى قالبى تماثل إثنين .**

قالب بيانات2 | قالب بيانات1

القرص - 1 | القرص - 2 | القرص - 3 | القرص - 4 | القرص - 5 | القرص - 6

الشكل (9.16) - المستوى–6 (RAID-6) – تكرار بحفظ شفرة التحكم بالأخطاء (خانات تماثل متداخلة – بمستوى قالب – موزع – مزدوج)

مقارنة مختصرة للمستويات السبعة من نظام الصفوف المتكررة من الأقراص المستقلة وتشمل المميزات والعيوب ومجالات الإستخدام يبينها الجدول (9.4) حيث يُعرض كل مستوى مع ذكر ميزاته وعيوبه و إستخداماته .

الجدول (9.4 – أ) - ملخص لمقارنة المستويات السبعة من نظام الصفوف المتكررة من الأقراص المستقلة (RAID)

المستوى	المميزات	العيوب	الإستخدام
0	أداء الإدخال/الإخراج تحسن كثيرا وذلك بتوزيع حمل الإدخال/الإخراج على عدة قنوات و مشغلات (الاقراص). لا يتضمن تعقيدات حساب التماثل. تصميم مبسط. سهل الإنجاز.	الفشل فى مشغل قرص واحد ينتج عنه فقدان كل البيانات فى الصفوف	أنتاج وإعداد الفيديو. أخراج الصور. لتطبيقات تتطلب معدل بيانات كبير.
1	تكرار للبيانات يصل الى 100% بحيث لا توجد حاجة لأعادة بناء البيانات فى حالة فشل القرص ، تنسخ البيانات من القرص المكرر. فى بعض الحالات (RAID-0) يمكنه تحمل فشل عدة اقراص بالتوازى. من أبسط أنظمة (RAID) الجزئية للتخزين فى التصميم.	الأعلى نفقة من حيث الاقراص فى جميع انواع انظمة (RAID) – غير عملى	الأنظمة المحاسبية. المالية. التمويل. أى تطبيق يتطلب أنظمة ذات جهوزية عالية ومتوفر دائما.
2	يمكنه نقل البيانات بمعدل عالى جداً. مع معدل نقل عالى نسبة البيانات الى شفرة تصحيح البيانات تتحسن. نسبيا تصميم المتحكم بسيط مقارنة بانظمة (RAID) 3 و 4 و5 .	نسبة شفرة تصحيح البيانات بالقرص الى بيانات القرص عالية جدا مع حجم كلمة صغير – غير عملى.	غير تجارى تم إنجازه ولكنه لم يسوق.
3	معدل نقل بيانات القراءة/الكتابة عالى جدا. فشل القرص له تأثير عالى على الخرج. نسبة شفرة تصحيح البيانات بالقرص الى بيانات القرص منخفضة – كفاءة عالية.	فى أحسن الأحوال معدل العمل مساوى لقرص فردى. تصميم المتحكم متوسط التعقيد.	أنتاج الفيديو والبث المباشر. أخراج الصور. أخراج الفيديو. أى تطبيقات تحتاج خرج عالى.

الجدول (9.4 – ب) - ملخص لمقارنة المستويات السبعة من نظام الصفوف المتكررة من الأقراص المستقلة (RAID)

المستوى	المميزات	العيوب	الإستخدام
4	ذو معدل عمليات قراءة للبيانات عالى جدا. نسبة شفرة تصحيح البيانات بالقرص الى بيانات القرص منخفضة – كفاءة عالية	تصميم المتحكم معقد. ذو معدل عمليات كتابة للبيانات سيئ. فى حالة فشل قرص يصعب أعادة بناء البيانات.	غير تجارى تم أنجازه ولكنه لم يسوق.
5	الأعلى من حيث معدل عمليات قراءة البيانات. نسبة شفرة تصحيح البيانات بالقرص الى بيانات القرص منخفضة – كفاءة عالية.	الأعقد من حيث تصميم المتحكم. فى حالة فشل قرص يصعب أعادة بناء البيانات.	الخوادم وانظمة الملفات، خوادم قواعد البيانات، خوادم البريد الاليكترونى ،والأخبار وشبكة المعلومات ، الخوادم المحلية. أكثر مستويات (RAID) استخداما.
6	يوفر أعلى مستوى تحمل لخطأ بيانات ويمكنه أن يتغلب على فشل متوازى فى عدة مشغلات.	تصميم المتحكم عالى التعقيد. التعقيد فى حساب التماثل عالى جداً.	حل نموذجى للأنظمة الخاصة بالمهام الحرجة.

9.3 سواقة الحالة الصلبة (SSD)

من أهم التطورات الأخيرة في عمارة نظام الحاسب هو الإستخدام المتزايد لسواقات الحالة الصلبة (Solid State Drive-SSD) لمساندة أو حتى أستبدال سواقات الأقراص الصلبة (HDD) وذلك بحيث تستخدم كذاكرة ثانوية داخلية وخارجية . ويشير مصطلح الحالة الصلبة إلى دوائر إلكترونية بنيت بعناصر شبه موصله ،

فسواقة الحالة الصلبة هى جهاز ذاكرة مصنوع من مكونات الحالة الصلبة التي يُمكن إستخدامها كبديل لسواقة القرص الصلب (HDD) . وحاليا سواقات الحالة الصلبة متوفرة فى الأسواق وكذلك يشيع الأن إستخدام نوع من الذاكرة الالكترونية الشبه موصلة تعرف بالذاكرة الوميضية (Flash Memory) .

وفي هذا القسم سنعرض أولاً مقدمة للذاكرة الوميضية وبعد ذلك ننظر في إستخدامها كنوع من سواقات الحالة الصلبة .

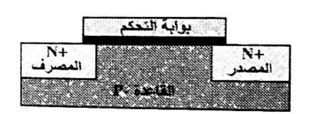
9.3.1 الذاكرة الوميضية

الذاكرة الوميضية هى نوع من الذاكرة الالكترونية الشبه موصلة وهى موجودة منذ عدة سنوات ، وتستخدم في العديد من المنتجات الإلكترونية الإستهلاكية بما في ذلك الهواتف الذكية وأجهزة نظام تحديد المواقع العالمي (GPS) ومشغلات الموسيقي (MP3) ، والكاميرات الرقمية ، وأجهزة الناقل التسلسلي العام (USB) . وفي السنوات الأخيرة تطورت تكلفة وأداء الذاكرة الوميضية لدرجة أنه اصبح من الممكن إستخدامها كبديل لسواقات الأقراص الصلبة (HDD) .

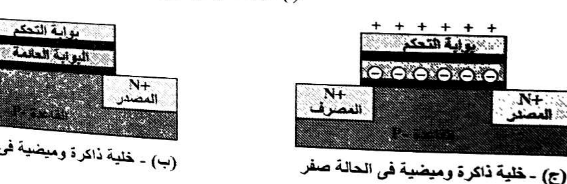
ويوضح الشكل (9.17) طريقة العمل الأساسية للذاكرة الوميضية ، وعلى سبيل التوضيح الشكل (9.17 – أ) يصور كيفية عمل الترانزستور ، **فالترانزستور مصنوع من عناصر تستخدم خصائص أشباه الموصلات بحيث أن تطبيق جهد صغير على البوابة يُمكِن من التحكم في تدفق تيار كبير مابين المَصدر والمُصَرف.**

في خلية الذاكرة الوميضية تُضاف بوابة ثانية إلى الترانزستور وتسمى بالبوابة العائمة وذلك لأنها معزولة بطبقة أُكسيد رقيقة . بدايةً ، لا تتدخل البوابة العائمة فى عمل الترانزستور (الشكل 9.17 – ب) ، وفي هذه الحالة تمثل الخلية الثنائي 1 ،

وعند تطبيق جهد كبير عبر طبقة الأكسيد يسبب ذلك أن الإلكترونات تخترقها **وتُصبح محصورة** على البوابة العائمة وتظل كذلك حتى ولو تم فصل الطاقة **الكهربية (الشكل** 9.17 – ج) ، وفي هذه الحالة تمثل الخلية الثنائي 0 . وحالة الخلية **يمكن قراءتها** بإستخدام دوائر خارجية تختبر ما إذا كان الترانزستور يعمل أم لا ، **وتطبيق جهد كبير** في الاتجاه المعاكس يزيل الإلكترونات من البوابة العائمة وتعود **إلى حالة تمثيل الثنائي** 0 .



(أ) - بنية الترانزستور



(ب) - خلية ذاكرة وميضية فى الحالة واحد

(ج) - خلية ذاكرة وميضية فى الحالة صفر

الشكل (9.17) –عمل الذاكرة الوميضية

هناك نوعان مميزان من الذاكرة الوميضية ويشار اليهما بمنطق **نفي-الجمع (NOR) و منطق نفي-الضرب** (NAND) . ففي الذاكرة الوميضية من نوع **منطق نفي-الجمع** (NOR) الوحدة الأساسية في التواصل هى خانة ثنائية ، **وتنظيمها المنطقي يشابه منطق دائرة نفي-الجمع** المنطقية (NOR) . والذاكرة **الوميضية من نوع منطق نفي-الضرب** (NAND) الوحدة الأساسية في التواصل

هي 16 أو 32 خانة ثنائية وتنظيمها المنطقي يشابه منطق دائرة **نفي-الضرب** المنطقية (NAND) .

تُوفر الذاكرة الوميضية من نوع منطق نفي-الجمع (NOR) الوصول العشوائي للبيانات وبسرعة عالية ، ويمكنها قراءة وكتابة البيانات إلى مواقع محددة ، وكذلك يمكنها التأشير وإسترجاع البيانات بالخانة . وتستخدم الذاكرة الوميضية من نوع منطق نفي-الجمع (NOR) فى تخزين برنامج نظام تشغيل الهاتف الخليوي وكذلك فى تخزين برنامج نظام البدء (BIOS) والذى يعمل عند بدء تشغيل أجهزة الحاسب المعتمدة على نظام ويندوز .

وأما الذاكرة الوميضية من نوع منطق نفي-الضرب (NAND) فهى تقرأ وتكتب فى قوالب صغيرة وهى مستخدمة فى السواقات الوميضية بالناقل التسلسلي العام (USB) وبطاقات الذاكرة (في الكاميرات الرقمية ومشغلات الموسيقى ، الخ) وسواقات الحالة الصلبة ، وتوفر الذاكرة الوميضية من نوع منطق نفي-الضرب (NAND) كثافة بيانات أعلى من نوع منطق نفي-الجمع (NOR) وكذلك سرعة كتابة اكبر ، وفى الذاكرة الوميضية من نوع منطق نفي-الضرب (NAND) لا يتوفر الوصول العشوائي للبيانات بناءاً على عنوان خارجي (من ناقل العناوين) لذلك يجب قراءة البيانات على أساس قولبى (المعروف أيضا بإسم التواصل بالصفحة) حيث يَسع كل قالب من مئات الى الآلاف من الخانات الثنائية.

9.3.2 مقارنة مابين السواقة الوميضية والسواقة القرصية

كلما أنخفضت تكلفة سواقات الذاكرة الوميضية وزاد أدائها ، وزادت كثافة خانات البيانات المخزنة عليها أصبحت اكثر تنافسية مع سواقات الأقراص الصلبة ، ويبين الجدول (9.5) نموذج لمقارنة مابين الأثنين .

سواقات الذاكرة الوميضية لديها المزايا التالية على سواقات الأقراص الصلبة :

1. **معدل أداء عالي لعمليات الإدخال/الإخراج في الثانية** (IOPS) : وهذا **يزيد بشكل ملحوظ من** أداء أنظمة الإدخال/الإخراج الفرعية .
2. **المتانة** : أقل عرضة للصدمات والاهتزازات .
3. **عمر أطول** : سواقات الحالة الصلبة ليست عرضة للتأكل الميكانيكي .
4. **أستهلاك الطاقة منخفض** : سواقات الذاكرة الوميضية تستخدم ما لا يزيد عن 2.1 وات من الطاقة للسواقة ، وذلك أقل بكثير من سواقات الأقراص الصلبة ذات حجم مماثل .
5. **أكثر هدوءاً وأبرد في التشغيل** : وهي تتطلب مساحة أقل ، وتكاليف طاقة منخفضة ، ولذلك فهي صديقة للبيئة .
6. **معدل منخفض لزمن الوصول و زمن التأخير** : أكثر من 10 مرات **أسرع من سواقات الأقراص الصلبة .**

وحالياً ، سواقات الأقراص الصلبة تتمتع بميزة التكلفة المنخفضة للخانة وميزة **السعة الضخمة نسبة الى** سواقات الذاكرة الوميضية ، ولكن هذه الاختلافات تتقلص.

الجدول (9.5) - مقارنة مابين السواقة الوميضية وسواقة القرصية

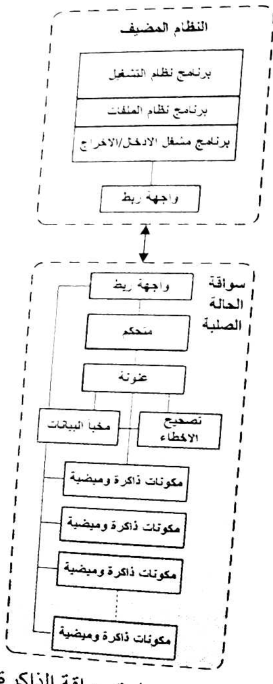
معدل الأداء / السواقة	سواقة قرصية	سواقة وميضية (NAND)
معدل الإدخال/الإخراج في الثانية	300	قراءة : 45,000 كتابة : 15,000
الخرج (ميغا ثمان / ثانية)	حتى 80	قراءة : 200+ كتابة : 100+
زمن الوصول العشوائي (ملي ثانية)	10 – 4	0.1
سعة التخزين	حتى 4 تيرا ثمان	حتى 256 غيغا ثمان

9.3.3 تنظيم سواقة الحالة الصلبة

يوضح الشكل (9.18) منظراً عاماً لمكونات النظام المعماري الشائع لأي نظام سواقة ذاكرة وميضية (SSD) . فعلى نظام التشغيل بالحاسب المضيف إستدعاء برنامج نظام الملفات للوصول إلى البيانات الموجودة على السواقة وبدوره برنامج نظام الملفات يستدعي برنامج مشغل الإدخال/الإخراج . وبرنامج مشغل الإدخال/الإخراج يوفر إمكانية وصول المضيف الى منتج معين من سواقات الذاكرة الوميضية ، ويشير عنصر واجهة الربط في الشكل (9.18) إلى الربط المادي والكهربائي مابين المعالج المضيف وجهاز سواقة الذاكرة الوميضية الطرفي ، فإذا كان الجهاز هو القرص الصلب الداخلي فواجهة الربط الشائعة هي رابط المكونات الطرفية السريع (PCIe) ، أما للأجهزة الخارجية فواجهة الربط الشائعة هي الناقل التسلسلي العام (USB) .

بالإضافة إلى واجهة الربط مع النظام المضيف فإن سواقة الذاكرة الوميضية تحتوي على العناصر التالية :

1. متحكم : يوفر مستوى الربط بجهاز سواقة الذاكرة الوميضية وأطار العمل.
2. عنونة : منطق مادي يؤدي وظيفة الأختيار مابين عدة سواقات ذاكرة وميضية .
3. مخبأ البيانات : وحدة ذاكرة عشوائية ذات سرعة عالية تستخدم لتسريع المقارنة وزيادة سرعة نقل البيانات .
4. **تصحيح الأخطاء : منطق مادي لأكتشاف الأخطاء وتصحيحها .**
5. **مكونات الذاكرة الوميضية : شرائح منطق نفي-الضرب (NAND) المكونة لخلايا الحفظ في الذاكرة الوميضية .**



الشكل (9.18) –عمارة سواقة الذاكرة الوميضية

9.4 الذاكرة الضوئية

القرص المدمج في عام 1983 قُدم منتج من أنجح المنتجات الاستهلاكية : القرص المدمج (Compact Disk-CD) للنظام السمعي الرقمي . القرص المدمج هو قرص غير قابل للمسح يمكنه تخزين أكثر من 60 دقيقة من المعلومات الصوتية على وجه (سطح) واحد . النجاح التجاري الضخم للقرص المدمج مَكن من تطوير تقنية

للتخزين الضوئي الرقمي على القرص بتكلفة منخفضة ، وقد أحدث هذا ثورة فى تخزين البيانات على الحاسب . والجدول (9.6) يقدم مجموعة متنوعة من أنظمة الأقراص الضوئية .

الجدول (9.6) - أنواع القرص الضوئى

القرص المدمج (CD):
قرص مدمج غير قابل للمسح لحفظ البيانات السمعية رقميا . قياسيا فالقرص بقطر 12 سم ، ويسع حوالى 60 دقيقة بدون انقطاع

القرص المدمج – للقراءة فقط (CD-ROM):
قرص مدمج غير قابل للمسح لحفظ بيانات حاسوبية . قياسيا فالقرص بقطر 12 سم ، ويسع حوالى 650 ميغا ثمان

القرص المدمج – قابل للتسجيل (CD-R):
مثل القرص المدمج – للقراءة فقط ولكن المستخدم يمكنه الكتابة عليه مرة واحدة فقط .

القرص المدمج – قابل للكتابة (CD-RW):
مثل القرص – للقراءة فقط ولكن يمكن للمستخدم مسحه ، والكتابة علية عدة مرات .

القرص المتعدد الاستخدام (DVD):
تقنية تسمح بانتاج تمثيل رقمى مضغوط لبيانات فيديو ، وكذلك الإحجام الكبيرة من البيانات الرقمية . تستخدم أقراص بقطرين 8 سم و 12 سم ، وبوجهين مزدوجين وبسعة تصل الى 17 غيغا ثمان . النوع الأساسى منها هو للقراءة فقط .

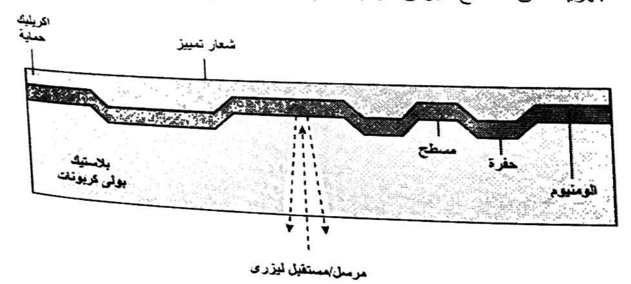
القرص المتعدد الاستخدام – قابل للتسجيل (DVD-R):
مثل القرص المتعدد الإستخدام- للقراءة فقط ، ولكن المستخدم يمكنه الكتابة مرة واحدة فقط ، ويستخدم وجه واحد فقط .

القرص المتعدد الاستخدام – قابل للكتابة (DVD-RW):
مثل القرص المتعدد الإستخدام- للقراءة فقط ، ولكن يمكن للمستخدم مسحه ، والكتابة عليه عدة مرات ، ويستخدم وجه واحد فقط .

القرص المتعدد الاستخدام ذو الشعاع الازرق (Blue-Ray DVD):
قرص فيديو عالى الوضوح ويوفر سعة تخزينية أكبر من القرص المتعدد الإستخدام ، وذلك بإستخدام ليزر 405 نانوميتر . قرص بطبقة واحدة و بوجه فردى يمكنه أن يحفظ 25 غيغا ثمان .

9.4.1 القرص المدمج (Compact Disk-CD)

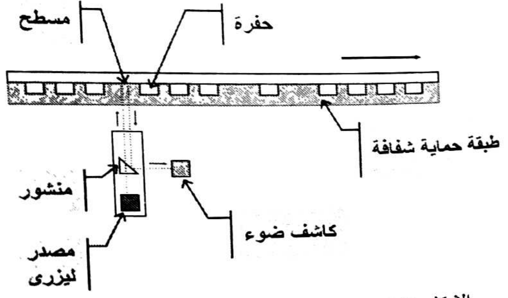
كل من القرص المدمج السمعي و القرص المدمج - للقراءة فقط (CD-ROM) يشتركان فى تقنية متشابهة ، والفرق الرئيسي هو أن القرص المدمج - للقراءة فقط (CD-ROM) أكثر صلابة وله آلية لتصحيح الاخطاء للتأكد من أن البيانات تم نقلها بشكل صحيح من القرص إلى جهاز الحاسب ، ويتم تصنيع كلا النوعين من الأقراص بنفس الطريقة ، ويصنع القرص من **مادة صمغية** مثل البولى كربونات. المعلومات تسجل رقميا (موسيقى أو بيانات حاسوبية) على شكل سلسلة من الحفر المجهرية على السطح البولى كربونات (الشكل 9.19) .



الشكل (9.19) – طريقة عمل القرص المدمج

يتم أسترداد المعلومات من القرص المدمج الصوتى أو القرص المدمج - للقراءة فقط بواسطة شعاع ليزر منخفضة الطاقة بمشغل القرص الضوئي . شعاع الليزر يضيء من خلال مادة البولي الشفافة أثناء دوران محرك القرص ، وشدة الضوء الليزرى المنعكس تتغير كلما قابلت حفرة . وعلى وجه التحديد ، فإذا سقط شعاع الليزر علي حفرة سطحها خشن بعض الشيء ، فإن الضوء سيتشتت وسينعكس

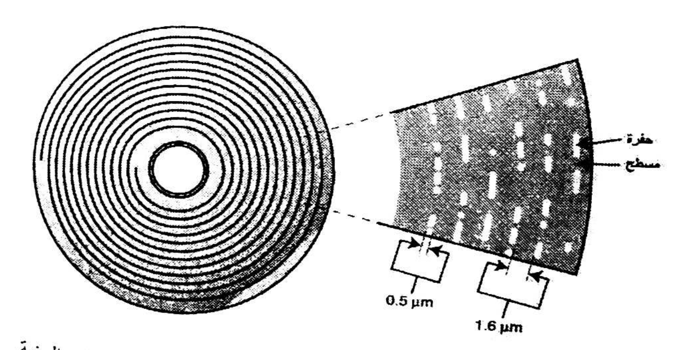
بكثافة منخفضة إلى المصدر . ويطلق على المناطق الواقعة بين الحفر ***مسطحات*** ، والمسطح عبارة عن سطح أملس يعكس الضوء الليزرى بكثافة أعلى . **الأنتقال** مابين الحفر والمسطحات يتم الكشف عنه من قبل حساس ضوئى من ثم **يتم تحويله** إلى إشارة رقمية ، والحساس يختبر السطح في فترات منتظمة . **بداية أو نهاية** حفرة تمثل (1) ، وعندما لا يحدث أي تغيير في الارتفاع مابين الفترات **يتم تسجيل** (0) ، وكما هو موضح فى الشكل (9.20) .



الشكل (9.20) – عملية القراءة من القرص المدمج

ولتحقيق مزيد من السعة فإن الأقراص المدمجة لا تنظم المعلومات على مسارات متحدة المركز ، وبدلا من ذلك ، فالقرص يحتوي على مسار لولبى واحد يبدأ بالقرب من المركز ويدور لولبيا للخارج إلى الحافة الخارجية للقرص ، والقطاعات القريبة للخارج هي بنفس طول التى بالقرب من الداخل (المركز) ، وبالتالي فإن المعلومات معبأة بالتساوي على القرص في مقاطع بنفس الحجم ويتم فحصها بنفس

المعدل عن طريق تدوير القرص بسرعة متغيرة . تتم قراءة الحفر من قبل الليزر **في سرعة خطية ثابتة** ، فالقرص يدور ببطء أكثر بالقرب من الحافة الخارجية **منها بالقرب** من المركز ، وهكذا فإن سعة المسار وتأخير التدوير تزداد للمواقع **بالقرب** من الحافة الخارجية للقرص . والشكل (9.21) يوضح تنظيم المسارات **على القرص** بحيث تصل سعة البيانات على الأقراص المدمجة لحوالي 680 ميغا ثمان .



الشكل (9.21) – المسارات اللولبية على القرص المدمج والمسافات البينية

يتم تنظيم البيانات على القرص المدمج كسلسلة من القوالب ، والشكل (9.22) **يظهر التنسيق النموذجيّ** للقالب ، ويتكون من الحقول التالية : -

- **المزامنة** (Sync) : حقل المزامنة يحدد بداية القالب ، ويتكون من 8-خانات من الأصفار $_{16}$(00) ، و 10 ثمان من الآحاد $_{16}$(FF ... FF) ، و8-خانات من الأصفار $_{16}$(00) .

- **المقدمة** (Header) : المقدمة تحتوي على عنوان القالب وثمان يبين وضع التخزين على القالب (Mode) . فالوضع (0) يبين أن حقل البيانات فارغ ؛ (1) يحدد استخدام 2048 ثمان للبيانات ورمز تصحيح الأخطاء كبنية للقالب ؛ الوضع (2) يحدد بنية القالب مكونة من 2336 ثمان للبيانات بدون رمز تصحيح الأخطاء .
- **البيانات** (Data) : بيانات المستخدم .
- **مساعدة** (Auxiliary) : في الوضع (2) هو عبارة عن بيانات إضافية للمستخدم ، وفي الوضع (1) هو عبارة عن رمز لتصحيح الأخطاء (ECC) من 288 ثمان .

00	FF ... FF	00	MIN	SEC	Sector	Mode	Data	Layered ECC
12 bytes مزامنة			4 bytes مقدمة				2048 bytes البيانات	288 bytes مساعدة
2352 bytes								

الشكل (9.22) - تنسيق القالب في القرص المدمج

ومع إستخدام السرعة الخطية الثابتة فإن الوصول العشوائي يصبح أكثر صعوبة ، بحيث أن تحديد عنوان معين ينطوي على تحريك الرأس إلى منطقة عامة ، ثم تعديل سرعة التدوير لقراءة العنوان ومن ثم إجراء تعديلات طفيفة لإيجاد والتواصل مع قطاع محدد .

الأقراص المدمجة مناسبة لتوزيع كميات كبيرة من البيانات على عدد كبير من المستخدمين ، وذلك بسبب تكلفة عملية الكتابة الأولية ، ولذلك فهي ليست مناسبة

للتطبيقات الفردية . مقارنة مع الأقراص المغناطيسية التقليدية ، فإن الأقراص **المدمجة لها ميزتان** :-

- القرص المدمج بالمعلومات المخزنة عليه يمكن نسخها بالجملة بتكلفة زهيدة بعكس القرص المغناطيسي . فقاعدة البيانات على القرص المغناطيسي يعاد إنتاجها عن طريق نسخ قرص واحد كل مرة باستخدام محركي أقراص .
- القرص الضوئي متحرك (قابل للإزالة/الفك) ، وهذا يسمح بإستخدام القرص نفسه كأرشيف تخزين ، فى حين أن أكثر الأقراص المغناطيسية هي ثابتة (غير قابلة للفك) . المعلومات على القرص المغناطيسي الثابت (الغير قابل للإزالة/الفك) يجب أولاً أن تنسخ على وسيلة تخزين أخرى قبل أن يستخدم مشغل القرص من جديد لتخزين معلومات جديدة .

عيوب القرص المدمج هي :-

- للقراءة فقط ولا يمكن تعديلها .
- زمن الوصول أطول منه عن محرك الأقراص المغناطيسية ، ويصل إلى نصف ثانية .

9.4.2 القرص المدمج القابل للتسجيل (CD-R)

ولأستيعاب التطبيقات التي تحتاج لنسخ مجموعة من البيانات مرة واحدة أو عدد محدود من المرات تم تطوير قرص مدمج للكتابة مرة واحدة والقراءة أكثر من مرة والمعروف باسم القرص المدمج القابل للتسجيل (CD-R) . ففى القرص المدمج القابل للتسجيل يتم إعداد القرص بحيث يمكن الكتابة عليه لاحقاً مرة واحدة بواسطة شعاع ليزرى محدود الشدة ، وبهذا يمكن للمستخدم الكتابة مرة واحدة و القراءة

المتعددة من القرص ، وذلك يتم بإستخدام وحدة تحكم بالقرص إلى حد ما **أكثر تكلفة** منها فى القرص المدمج - للقراءة فقط .

وسط القرص المدمج القابل للتسجيل يشبهه فى القرص المدمج - **للقراءة فقط ولكنه** ليس متطابق ، وسط القرص المدمج القابل للتسجيل **يشمل طبقة صبغية** ، **ويتم** إستخدام الصبغة لتغيير الإنعكاسية ويتم تفعيلها من خلال **ليزرعالي الشدة** . والقرص يمكن إستخدامه على مشغل الأقراص القابلة للتسجيل **أو مشغل الأقراص** المدمجة - للقراءة فقط .

9.4.3 القرص المدمج القابل لإعادة الكتابة (CD-RW)

تقنية القرص الضوئى المدمج القابل لإعادة الكتابة **(CD-RW) تمكن من إعادة** الكتابة عليه مرات عديدة وكما هو الحال مع القرص المغناطيسي . **التقنية الضوئية** التي مكنت من إعادة الكتابة تسمى تقنية **تغيير الطور (Phase change)** . فالقرص ذو الطور المتغير يستخدم مواد لها **إنعكاسين مختلفين في حالتى طور مختلفتين** . هناك حالة غير متبلور للمادة بحيث أن جزيئاتها **تُظْهر سلوك عشوائي** مما يعكس الضوء بشكل سيئ ، وحالة آخرى بلورية حيث **تشكل سطحاً أملساً مما** يعكس الضوء جيداً . ويمكن لشعاع من ضوء الليزر **أن يغير حالة المادة من حالة** إلى آخرى . والعيب الرئيسي للأقراص الضوئية متغيرة الطور **هو أن المواد تفقد** خصائصها المرغوب فيها بالتدريج وبشكل دائم . **المواد الحالية يمكن إستخدامها** لما بين 500000 و 1000000 **دورة إعادة كتابة** .

القرص المدمج القابل لإعادة الكتابة لديه ميزة إمكانية تكرار الكتابة بحيث يمكن إعتباره وحدة تخزين ثانوية حقيقية ، وعلى هذا النحو فإنه ينافس القرص المغناطيسي .

9.4.4 الأقراص الرقمية المتعددة الإستخدامات (DVD)

الأقراص الرقمية المتعددة الإستخدامات حلت محل أشرطة الفيديو المستخدمة في مسجلات الفيديو (أجهزة تسجيل فيديو المنزلية) واستبدلت الأقراص المدمجة في أجهزة الحاسوب الشخصية والخوادم . الأقراص الرقمية المتعددة الإستخدامات أدخلت الفيديو إلى العصر الرقمي بحيث قدمت أفلام بجودة صورة رائعة ، ويمكن الوصول إليها بشكل عشوائي مثلما في الأقراص المدمجة السمعية والتي يمكن لأجهزة الأقراص الرقمية المتعددة الاستخدامات أن تشغلها أيضا . مع الأقراص الرقمية المتعددة الإستخدامات ذات القدرة التخزينية الضخمة والجودة العالية أصبحت الألعاب الحاسوبية برامج أكثر واقعية والبرامج التعليمية تضمنت مزيداً من الصور والفيديو .

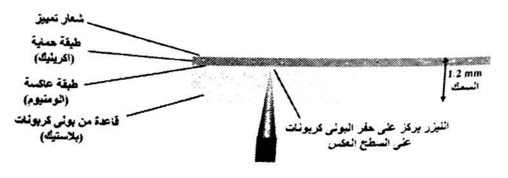
السعة الكبيرة للأقراص الرقمية المتعددة الإستخدامات (DVD) تعود إلى ثلاثة أختلافات عن الأقراص المدمجة (الشكل 9.23) :

1. الخانات الثنائية مخزنة على نحو اكثر تراصا . فالتباعد بين الحلقات اللولبية على القرص المدمج (CD) هو 1.6 μm والحد الأدنى للمسافة بين الحفر على طول اللولب هو 0،834 μm . في حين أن الأقراص الرقمية المتعددة الإستخدامات تستخدم ليزر بطول موجي أقصر وتحقق تباعد للحلقة اللولبية 0.74 μm والحد الأدنى للمسافة بين الحفر 0.4 μm، ونتيجة لهذه التحسينات زادت السعة إلى حوالي 4.7 غيغا ثمان .
2. الأقراص الرقمية المتعددة الإستخدامات تستخدم طبقة ثانية من الحفر والمسطحات فوق الطبقة الأولى . فالأقراص الرقمية المتعددة الإستخدامات المزدوجة الطبقة تحتوي على طبقة شبه-عاكسة فوق الطبقة العاكسة وبالضبط البؤرى يمكن للشعاع الليزرى في مشغلات هذه

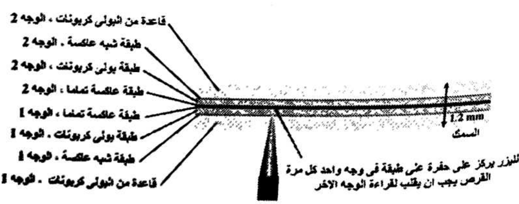
الأقراص من قراءة كل طبقة على حدة . إن أستعمال هذه التقنية ضاعف تقريبا من السعة التخزينية للقرص الى 8.5 غيغا ثمان تقريبا ، وأنخفاض أنعكاسية الطبقة الثانية يحد من الطاقة التخزينية بحيث لا يتم التوصل إلى مضاعفة كاملة .

3. يمكن الأقراص الرقمية المتعددة الاستخدامات – للقراءة فقط أن تكون مزدوجة الوجه في حين أن في القرص المدمج يتم تسجيل البيانات على وجه واحد فقط من القرص ، وهذا يزيد من السعة لتصل حتى 17 غيغا ثمان .

للأقراص الرقمية المتعددة الاستخدامات عدة أنواع ، فمنها قابل للكتابة وكذلك للقراءة فقط (الجدول 9.6).



(أ) قرص مدمج بسعة 682 ميغا ثمان



(ب) قرص (DVD) ، مزدوج الطبقة ، مزدوج الوجه ،بسعة 17 جيجا ثمان

الشكل (9.23) - الأقراص الرقمية المتعددة الاستخدامات والأقراص المدمجة

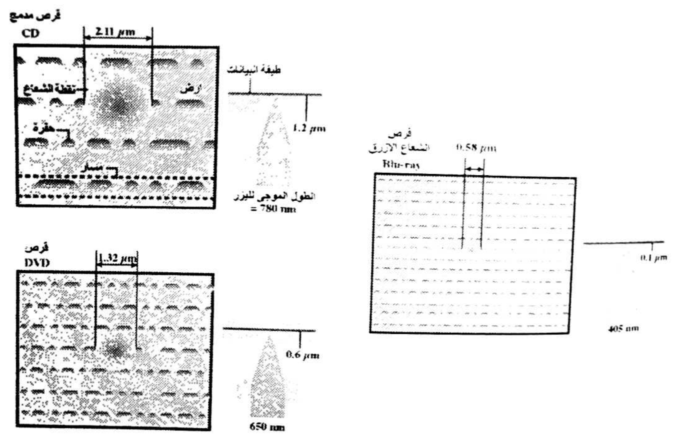
9.4.5 الأقراص الضوئية عالية الوضوح (HD)

تم تصميم الأقراص الضوئية العالية الوضوح (HD) لتخزين فيديو عالي الوضوح **ولتوفير** قدر أكبر من السعة التخزينية مقارنة بالأقراص الرقمية المتعددة **الأستخدامات** . لتحقيق كثافة تخزينية عالية تم أستخدام ليزر بطول موجي أقصر **في النطاق البنفسجي-الأزرق** ، ونتج عن ذلك أن حفر البيانات التي تشكل الخانات **الرقمية** 1 **و** 0 **هي** أصغر فى الأقراص الضوئية العالية الوضوح منها عن **الأقراص** الرقمية المتعددة الاستخدامات وذلك بسبب قصر الطول الموجي لليزر .

نوعين من الأقراص والتقنيات تتنافسان لنيل قبول السوق :

الأقراص الضوئية العالية الوضوح - متعددة الأستخدامات (HD DVD) **والأقراص** الرقمية المتعددة الأستخدامات - زرقاء الشعاع (Blue-ray DVD) ، **وفي نهاية المطاف** حققت تقنية الشعاع الأزرق الهيمنة على السوق حيث مكنت **من تخزين** 15 غيغا ثمان على طبقة واحدة وعلى وجه واحد للقرص .

فى تقنية الشعاع الأزرق موضع طبقة البيانات على القرص قريب جداً إلى الليزر **(كما هو موضح** فى الجانب الأيمن فى كل رسم بالشكل 9.24) ، وهذا مكن من **تضييق التركيز البؤرى وتقليل الأنحراف مما سمح بتصغير الحفر والمسارات.** **تقنية الشعاع الازرق يمكنها تخزين 25 غيغا ثمان على طبقة واحدة ، وتوجد حالياً ثلاثة إصدارات متوفرة منها :** للقراءة فقط (BD-ROM) ، **والتسجيل مرة واحدة (BD-R) ، والقابلة لإعادة التسجيل (BD-RE) .**



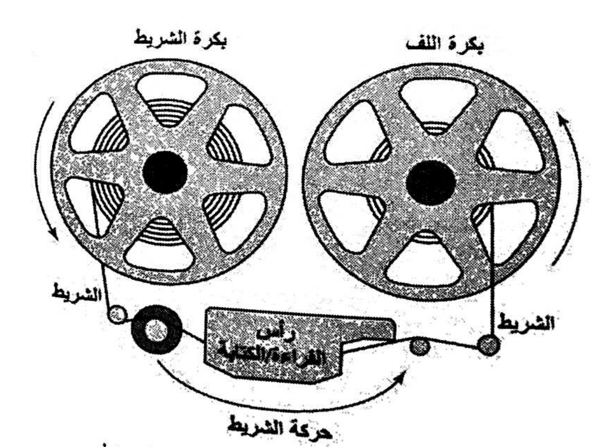
الشكل (9.24) – خصائص أقراص الذاكرة الضوئية

9.5 الشريط الممغنط

يستخدم نظام الشريط الممغنط نفس تقنيات القراءة والتسجيل فى نظم القرص المغناطيسي . فوسط التسجيل هو شريط بوليستر مرن مطلى **بمواد قابلة للمغنطة،** والطلاء قد يكون بجزيئات من معدن نقي بترابط الخاص أو **غشاء معدنى مُطلي** بخارياً .

الشريط ومحركه هو مشابه إلى نظام جهاز التسجيل الصوتى المنزلى ، وعرض الشريط يتراوح من 0.38 سم إلى 1.27 سم . اليوم ، جميع الأشرطة الممغنطة توضع في خراطيش أو فى بكرات ولكل منها نظام تشغيل خاص بها ، وكما هو موضح فى الشكل (9.25) والشكل (9.26) .

الشكل (9.25) – خرطوشة الشريط الممغنط



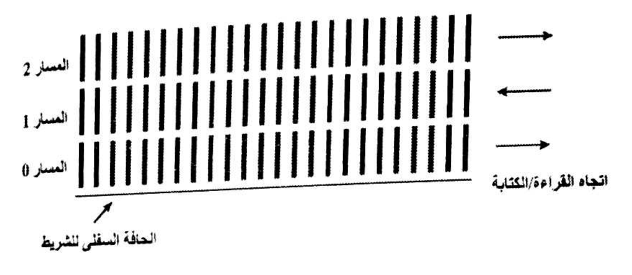
الشكل (9.26) – منظومة بكرة الشريط الممغنط

تنظم البيانات على الشريط فى عدد من المسارات المتوازية تُشغل طولياً . ومعظم النظم الحديثة تستخدم التسجيل التسلسلى بحيث توضع البيانات كخانات متتابعة

على طول كل مسار ، وكما هو فى الأقراص المغناطيسية تتم قراءة البيانات وكتابتها في قوالب متجاورة تدعى **السجلات** . يتم فصل القوالب على الشريط بفجوات تدعى **بينية السجل** ويتم تنسيق الشريط بشكل يساعد في تحديد مكان السجلات .

يشار إلى تقنيات التسجيل التقليدية المستخدمة في الأشرطة التسلسلية **بالتسجيل الإنعطافى** . وفي هذه التقنية ، عندما يتم تسجيل بيانات فإن المجموعة الأولى من البيانات يتم تسجيلها على طول الشريط كله ، وعند الوصول الى نهاية الشريط يتم إعادة موضع الرأس لتسجيل مسار جديد وعلى طول الشريط مرة أخرى ولكن هذه المرة في الإتجاه المعاكس ، وتستمر هذه العملية ذهابا وإيابا حتى يمتلئ الشريط بالكامل (الشكل 9.27 – أ) . ولزيادة السرعة فإن رأس القراءة-الكتابة قادر على قراءة وكتابة عدد من المسارات المتاخمة في وقت واحد (عادة 2-8 المسارات) . البيانات يتم تسجيلها تسلسليا على طول كل مسار ، ولكن القوالب المتسلسلة تخزن على مسارات متاخمة وكما هو موضح فى الشكل (9.27 – ب) .

مشغل الشريط هو جهاز وصول تتابعى ، فإذا تم وضع رأس الشريط على السجل-1 و نريد قراءة السجل-ن ، لابد من قراءة السجلات المادية من 1 إلى ن-1 قبله ، وإذا كان الوضع الحالي للرأس بعد السجل المطلوب ، **لا بد من ترجيع الشريط مسافة معينة (نقطة البداية) ونبدأ القراءة للإمام ، فالشريط يتحرك أثناء عملية القراءة أو الكتابة فقط** .



(أ) القراءة و الكتابة بطريقة التسجيل الانعطافي

المسار 3	4	8	12	16	20
المسار 2	3	7	11	15	19
المسار 1	2	6	10	14	18
المسار 0	1	5	9	13	17

اتجاه حركة الشريط ←

(ب) موضع القوالب لنظام قراءة/كتابة 4 مسارات بالتوازى

الشكل (9.27) – المميزات النموذجية للشريط الممغنط

رغم أن الشريط الممغنط يعتبر أول نوع من الذاكرة الثانوية ، فإنه لا يزال يستخدم على نطاق واسع لأنه الأقل من حيث التكلفة ، وسرعته الابطأ في التسلسل الهرمي للذاكرة ، وتقنية الشريط المهيمنة حاليا هى نظام الخرطوشة المعروف بأسم الشريط الخطى-المفتوح (LTO) .

مصطلحات مهمة

Access Time	زمن الوصول
Transfer Time	زمن النقل
Track	مسار
Substrate	الركيزة
Striped Data	بيانات مجزئة
Serpentine Recording	التسجيل الانعطافى
Seek Time	زمن البحث
Sector	قطاع
Rotational Delay	تأخير الدوران
Removable Disk	قرص متحرك/متنقل
RAID	الصفوف المتكررة من الأقراص المستقلة
Platter	طبق
Pit	حفرة
Optical Memory	ذاكرة ضوئية
Nonremovable Disk	قرص ثابت
Multiple Zoned Recording	تسجيل متعدد المناطق
Movable-Head Disk	قرص برأس متحرك
Magnetoresistive	مقاومة مغنطيسية
Magnetic Tape	شريط ممغنط
Magnetic Disk	قرص مغنطيسي
Land	مسطح
Head	رأس
Gap	فجوة
Floppy Disk	قرص مرن
Fixed-Head Disk	قرص برأس ثابت
DVD-RW	القرص المتعدد الإستخدام – قابل للكتابة
DVD-R	القرص المتعدد الإستخدام – قابل للتسجيل
DVD-ROM	القرص المتعدد الإستخدام – للقراءة فقط
DVD	القرص المتعدد الإستخدام
Cylinder	أسطوانة
CD-RW	قرص مدمج – قابل للكتابة
CD-R	قرص مدمج – قابل للتسجيل
CD-ROM	قرص مدمج – للقراءة فقط
Compact Disk (CD)	قرص مدمج

الشعاع الأزرق	Blue-Ray
السرعة الخطية الثابتة	Constant Linear Velocity (CLV)
السرعة الزاوية الثابتة	Constant Angular Velocity (CAV)
الذاكرة الوميضية	Flash Memory
سواقة الحالة الصلبة	Solid State Drive (SSD)
سواقة القرص الصلب	Hard Disc Drive (HDD)
الناقل التسلسلي العام	Universal Serial Bus (USB)
نظام تحديد المواقع العالمي	Global Positioning System (GPS)
نسق تشفير سمعي	MP3
عملية إدخال/اخراج في الثانية	IOPS
ضريبة الكتابة	Write Penalty
تماثل	Parity
وضع	Mode
شفرة التحكم بالأخطاء	Error Control Code (ECC)
مساعد/مساند	Auxiliary
مقدمة	Header
الشريط الخطي-المفتوح	LTO
رابط المكونات الطرفية السريع	PCIe
متزامن	Synchoronus
تسجيل	Record
تسلسلي	Serial
الأقراص الضوئية عالية الوضوح	High Definition (HD)

أسئلة للمراجعة

1. فى الذاكرة الخارجية عرف مايلى : المسار والمقطع والأسطوانة و الرأس المتحرك و الثابتة والتخزين الأنعطافي في الشريط الممغنط ؟ .
2. ماهى أنواع وخصائص القرص؟ .
3. وضح آلية القراءة الكتابة من القرص المغناطيسى ؟ .
4. وضح العوامل التى تؤثر فى سرعة القرص المغناطيسى ؟ .
5. قرص مغناطيسى به 8 اسطح ذو وجهين ، كل سطح به 512 مسار و كل مسار به 64 مقطع . حجم المقطع 256 كيلو ثُمان ، ومتوسط زمن البحث 8 ملى ثانية وزمن التأخير 8.3 ملى ثانية ، ماهى سعة القرص وماهو متوسط زمن التواصل ؟ .
6. مالفرق مابين القرص المدمج (CD) والقرص متعدد الأستخدامات (DVD) وماهو تنسيق القالب على القرص المضغوط ؟ .
7. مالفرق بين السرعة الزاوية الثابتة (CAV) والسرعة الخطية الثابتة (CLV) وفيما تستعمل كل منهما ؟ ز
8. عرف ما المقصود بنظام الصفوف المتكررة من الأقرص المستقلة مع توضيح مستوياتها السبعة .
9. أفترض نظام الصفوف المتكررة من الأقرص المستقلة مكون من 4 مشغلات وكل منها بسعة 200 غيغا ثمان ، ماهى السعة المتوفرة للتخزين لكل من المستوى 0 ، و1 ، و3 ، و4 ، و5 ، و6 ؟ .
10. أفترض قرص مغناطيسي له 8 أوجه ، 512 مسار لكل سطح و 64 مقطع لكل مسار و سعة المقطع 1 كيلو ثمان . متوسط زمن البحث هو 8 ملى ثانية وزمن الأنتقال من مسار الى مسار 1.5 ملى ثانية ومشغل القرص يدور بسرعة 3600 لفة في الثانية ويمكن قراءة المسارات المتتالية في أسطوانة بدون تحريك الرأس .
   أ- ماهي السعة الأجمالية للقرص ؟ .
   ب- ماهو متوسط زمن الوصول ؟ .



الفصل العاشر

وحدات الإدخال/الإخراج

10- وحدات الإدخال/الإخراج

بالإضافة إلى وحدة المعالجة المركزية و وحدات الذاكرة (الداخلية و الخارجية) العنصر الأساسي الثالث فى نظام الحاسب هو مجموعة وحدات الإدخال/الإخراج ، فكل وحدة من وحدات الإدخال/الإخراج ترتبط بناقل النظام وتتحكم فى واحد أو أكثر من الأجهزة الطرفية . وحدة الإدخال/الإخراج ليست مجرد مجموعة من الروابط الميكانيكية التي تربط جهاز مع ناقل النظام ، ففي الواقع وحدة الإدخال/الإخراج تحتوي على منطق مادي لأداء وظيفة التواصل بين الطرفية أو الجهاز وبقية مكونات نظام الحاسب عبر الناقل .

من أسباب عدم توصيل الأجهزة الطرفية مباشرة إلى ناقل النظام ما يلي :

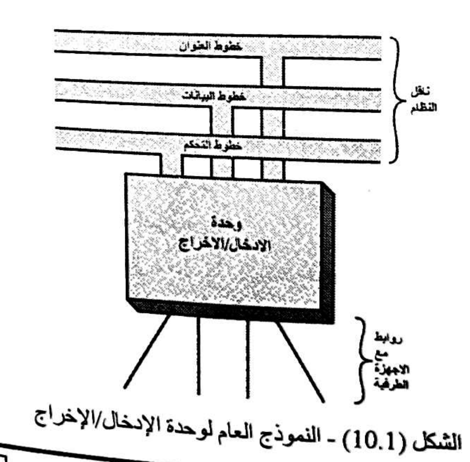
1. هناك مجموعة متنوعة و واسعة من الأجهزة الطرفية وبأساليب عمل مختلفة ، لذلك ، فإنه من غير العملي دمج المنطق المادى اللازم للتحكم فى هذه المجموعة المتنوعة من الأجهزة الملحقة (طرفيات) مع المعالج .
2. معدل نقل البيانات من الأجهزة الطرفية غالبا ما يكون أبطأ بكثير منه فى الذاكرة أو المعالج ، وبالتالي ، فإنه من غير العملي إستخدام نظام ناقل عالي السرعة للإتصال مباشرة مع الطرفية .
3. من ناحية أخرى ، فإن معدل نقل البيانات لبعض الأجهزة الطرفية هو أسرع من الذاكرة أو المعالج ، مرة أخرى ، فإن عدم التطابق يؤدي إلى عدم الكفاءة إذا لم تتم إدارته بشكل صحيح .

4. الملحقات غالبا ما تستخدم بيانات مختلفة الأشكال ، والأطوال ، والتنسيقات عن الحاسب المرفقة به .

بالتالي كانت هناك الحاجة الى وحدة الإدخال/الإخراج ، ولوحدة الإدخال/الإخراج وظيفتين رئيسيتين (الشكل 10.1) :

- الربط مع المعالج والذاكرة عن طريق ناقل النظام .
- الربط مع واحدة أو أكثر من الأجهزة الطرفية بوصلات بيانية خاصة .

نبدأ هذا الفصل بمناقشة وجيزة عن الأجهزة الخارجية ، تليها لمحة عامة عن بنية ووظيفة وحدة الإدخال/الإخراج ، ثم ننظر إلى الطرق المختلفة التي تُمكن وحدة الإدخال/الإخراج من تأدية وظيفتها بالتعاون مع المعالج والذاكرة : ربط مكونات النظام الأخرى مع وحدة الإدخال/الإخراج . وأخيرا ، ندرس الربط الخارجي مع وحدة الإدخال/الإخراج ومابين وحدة الإدخال/الإخراج والعالم الخارجي .



الشكل (10.1) - النموذج العام لوحدة الإدخال/الإخراج

10.1 الأجهزة الخارجية (الملحقات الطرفية)

يتم إنجاز عمليات الإدخال/الإخراج من خلال تشكيلة واسعة من الأجهزة الخارجية والتي توفر وسائل لتبادل البيانات بين المحيط الخارجي ونظام الحاسب الآلى . الاجهزة الخارجية تُرفق بالحاسب عن طريق رابط وحدة الإدخال/الإخراج (الشكل 10.1) ، ويستخدم الرابط لتبادل إشارات التحكم ، والحالة ، والبيانات بين وحدة الإدخال/الإخراج والجهاز الخارجي ، وغالباً ما يشار للجهاز الخارجي المتصل بوحدة الإدخال/الإخراج كجهاز ملحق أو ببساطة طرفية .

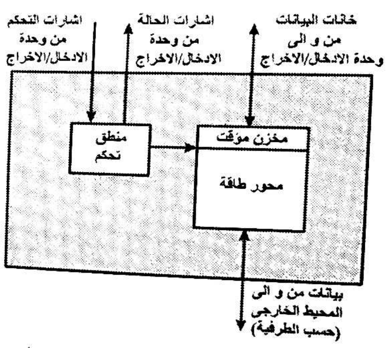
على نطاق واسع يمكننا تصنيف الأجهزة الخارجية إلى ثلاث فئات :

- المتعاملة مع الأنسان : مناسبة للتواصل مع مستخدم الحاسب .
- المتعاملة مع الآلة : مناسبة للتواصل مع المعدات و الأجهزة .
- الأتصالات : مناسبة للتواصل مع الأجهزة البعيدة .

أمثلة على الأجهزة المتعاملة مع الانسان هي طرفية عرض الفيديو (VDTs) والطابعات . وأمثلة على الأجهزة المتعاملة مع الآلة هي القرص المغناطيسي وأنظمة الشريط المغناطيسي ، وأجهزة الأستشعار والمشغلات الميكانيكية ، مثل تطبيقات الرجل الألى . ولاحظ أننا ننظر للقرص وأنظمة الشريط كأجهزة للإدخال/الإخراج في هذا الفصل بينما في الفصل (9) نظرنا اليها كأجهزة ذاكرة خارجية ، فمن وجهة نظر وظيفية هذه الأجهزة هي جزء من التسلسل الهرمي للذاكرة ، وقد تمت مناقشة إستخدامها في الفصل (9) ، ومن وجهة نظر هيكلية أو بنائية فإنه يتم التحكم في هذه الأجهزة عن طريق وحدات الإدخال/الإخراج وهذا ما سنتطرق إليه في هذا الفصل.

أجهزة الأتصالات تسمح لجهاز الحاسب بتبادل البيانات مع جهاز آلى بعيد ، والذى **قد يكون جهاز** يتعامل بشرياً مثل الطرفية ، أو جهاز مقروء آلياً ، أو حتى حاسب آخر .

بعبارات عامة جداً فإن طبيعة الجهاز خارجي يوضحها الشكل (10.2) . والربط مع وحدة الإدخال/الإخراج يتم في شكل إشارات التحكم ، وإشارات الحالة ، وإشارات البيانات .



الشكل (10.2) - الرسم تخطيطي لجهاز خارجي

إشارات التحكم تحدد الوظيفة التى سوف يؤديها الجهاز مثل إرسال البيانات إلى **وحدة الإدخال/الإخراج** (إدخال أو قراءة) ، أو قبول البيانات من وحدة **الإدخال/الإخراج** (إخراج أو الكتابة) ، أو تقرير حالة ، أو أجراء بعض وظائف التحكم الخاصة بالجهاز (على سبيل المثال تموضع رأس القرص) . **البيانات هى على شكل مجموعة من الخانات الثنائية** لإرسالها إلى أو إستقبالها من وحدة

الإدخال/الإخراج . إشارات الحالة تشير إلى حالة الجهاز ، مثل إشارة **مستعد/غير مستعد** وذلك لإظهار ما إذا كان الجهاز جاهز لنقل البيانات أم لا .

المنطق المادى للتحكم (**المُتحكّم**) المقرن مع الجهاز يتحكم في عمل الجهاز إستجابة لتوجيهات وحدة الإدخال/الإخراج . **مُحوّر الطاقة** (Transducer) يحول البيانات من شكل كهربائى إلى أشكال أخرى من الطاقة أثناء الإخراج ، ومن أشكال أخرى الى شكل كهربائى (إشارة كهربائية) أثناء الإدخال . وعادة ، يرتبط المُحَوّر مع وحدة تخزين مؤقت للتخزين المؤقت للبيانات التي يتم نقلها مابين وحدة الإدخال/الإخراج والمحيط الخارجي ، ومن الشائع أن يكون حجم المخزن مؤقت من 8 إلى 16 خانة .

سوف ندرس بإيجاز الأرتباط بين وحدة الإدخال/الإخراج والجهاز الخارجي لاحقاً. والأرتباط بين الجهاز الخارجي والمحيط الخارجى هو خارج نطاق هذا الكتاب ولكن سوف يتم التطرق لبعض الأمثلة الموجزة .

10.1.1 لوحة المفاتيح / المرقاب

الوسيلة الأكثر شيوعا للتفاعل بين الحاسب والمستخدم هو نسق **لوحة المفاتيح / شاشة العرض** . فالمستخدم يقوم بالإدخال من خلال لوحة المفاتيح ، **والبيانات المدخلة تنقل إلى الحاسب بحيث يمكن أيضا عرضها مباشرةً على الشاشة ، وإضافة إلى ذلك شاشة العرض تقوم بعرض البيانات المخرجة من الحاسب بعد المعالجة .** الوحدة الأساسية لتبادل البيانات هى الأحرف ، **وكل حرف يمثل برمز وهو عادة بطول 7 أو 8 خانات ثنائية** ، والرموز النصية **الأكثر إستخداما هي الأبجدية المرجعية الدولية** (IRA) . **ويتم تمثيل كل حرف في هذه المدونة بشفرة ثنائية**

وحيدة من 8 خانات ، وبالتالي فإنه يمكن تمثيل 128 حرفاً مختلفاً . الأحرف هي نوعين : أحرف طباعة وأحرف تحكم ، والأحرف القابلة للطباعة هي الأحرف الأبجدية والرقمية والخاصة التي يمكن طباعتها على الورق أو عرضها على شاشة العرض . بعض أحرف التحكم لها علاقة بالتحكم في الطباعة أو عرض الأحرف؛ مثال على ذلك إرجاع/إدخال ، وبعض أحرف التحكم الأخرى تهتم بإجراءات الإتصال .

وللإدخال من لوحة المفاتيح فإنه عندما يقوم المستخدم بخفض مفتاح (الضغط عليه) فأنه يولد إشارة الكترونية يتم تفسيرها من قبل المُحَور في لوحة المفاتيح وتُترجم إلى نمط من الخانات حسب المقابل لها فى المدونة الأبجدية المرجعية الدولية ، ومن ثم يرسل هذا النمط من الخانات الى وحدة الإدخال/الإخراج في الحاسب ، والحاسب يمكن أن يخزن النص بنفس الرمز . في الإخراج الرمز يُرسل من وحدة الإدخال/الإخراج الى الجهاز الخارجي ، والمُحَوّر في الجهاز يفسر الرمز ويرسل الإشارات الإلكترونية المناسبة لجهاز الإخراج إما لعرض الحرف المشار إليه أو أداء وظيفة تَحكم مطلوبة .

10.1.2 مشغل الأقراص

مشغل الأقراص يحتوي على الكترونيات لتبادل إشارات التحكم وإشارات الحالة وإشارات البيانات مع وحدة الإدخال/الإخراج بالإضافة إلى الكترونيات للتحكم في عمل آلية القراءة/الكتابة من على القرص .

في القرص ثابت الرأس ، المحوّر قادر على تحويل البيانات من صورتها كأنماط مغناطيسية على سطح القرص المتحرك الى خانات ثنائية للتخزين المؤقت على الجهاز (الشكل 10.2) . وبالنسبة للقرص المتحرك الرأس فإن المشغل يجب أن

يكون قادراً على تحريك ذراع القرص على سطح القرص شعاعياً للداخل أو للخارج .

10.2 وحدات الإدخال/الإخراج

في هذا القسم سوف نتطرق الى وظائف وبنية عمل وحدات الإدخال/الإخراج في نظام الحاسب الآلي .

10.2.1 وظيفة وحدة الإدخال/الإخراج

المهام أو المتطلبات الرئيسية لوحدة الإدخال/الإخراج يمكن تلخيصها فى التالي :

- التحكم والتزامن .
- الإتصال بالمعالج .
- الإتصال بالجهاز .
- التخزين المؤقت للبيانات .
- كشف الأخطاء .

خلال أي فترة من الزمن فإن المعالج قد يتواصل مع جهاز أو أكثر من الأجهزة الخارجية ويتم ذلك بأنماط عشوائية وحسب حاجة البرمجيات للإدخال او الإخراج. وبما أن الموارد الداخلية للنظام (مثل الذاكرة الرئيسية وناقل النظام) يتم تقاسمها ما بين عدة أنشطة من بينها التعامل مع البيانات ، ولذلك فإن من وظائف وحدة الإدخال/الإخراج التحكم و التزامن وذلك لتنسيق تدفق مرور البيانات بين الموارد الداخلية والأجهزة الخارجية . فعلى سبيل المثال ، التحكم فى نقل البيانات من جهاز خارجي إلى المعالج قد ينطوي على التسلسل التالي من الخطوات :

1. **المعالج يستجوب وحدة الإدخال/الإخراج للتحقق من حالة الجهاز المرفق (الجهاز المرتبط بالوحدة) .**
2. وحدة الإدخال/الإخراج ترد بحالة الجهاز المرفق .
3. إذا كان الجهاز شغالاً ومستعداً للإرسال فإن المعالج يطلب نقل البيانات عن طريق إرسال أمر إلى وحدة الإدخال/الإخراج .
4. وحدة الإدخال/الإخراج تتحصل على وحدة من البيانات (على سبيل المثال 8 أو 16 خانة) من الجهاز الخارجي .
5. يتم نقل البيانات من وحدة الإدخال/الإخراج إلى المعالج .

إذا كان النظام يستخدم الناقل فإن كل التفاعلات بين المعالج و وحدة **الإدخال/الإخراج** تنطوي على تحكيم أو أدارة السيطرة على الناقل (التحكم و تنسيق **إستغلال الناقل**) .

السيناريو المبسط السابق يوضح أيضا أنه يجب على وحدة الإدخال/الإخراج أن **تتواصل مع المعالج** ومع الجهاز الخارجي ، **والإتصالات بالمعالج** تنطوي على ما **يلي :**

- **حل شفرة الأمر** : وحدة الإدخال/الإخراج تتقبل الأوامر من المعالج ، **وعادة ما تُرسل هذه الإشارات** على متن ناقل التحكم . على سبيل المثال ، **وحدة الإدخال/الإخراج لمشغل الأقراص** قد تستقبل الأوامر التالية : قراءة **مقطع ، كتابة مقطع ، بحث عن رقم مسار** ، ومسح مُعرف سجل . **والأمرين الأخيرين يشملان معاملات يتم إرسالها على متن ناقل البيانات.**

- **البيانات** : يتم تبادل البيانات بين المعالج و وحدة الإدخال/الإخراج بواسطة ناقل البيانات .
- **تقرير الحالة** : لأن الأجهزة الطرفية بطيئة جداً ، لذلك من المهم معرفة ومتابعة حالة وحدة الإدخال/الإخراج . على سبيل المثال ، إذا تم الطلب من وحدة الإدخال/الإخراج إرسال بيانات إلى المعالج (قراءة) ، فربما لا تكون مستعدة لفعل ذلك لأنها لا تزال تعمل على أمر إدخال/إخراج سابق، فهذه الواقعة يمكن الإبلاغ عنها بإشارة حالة . إشارات الحالة الشائعة هي مشغول أو جاهز ، وقد يكون هناك أيضا إشارات للإبلاغ عن حالات أخطاء مختلفة .
- **التعرف على العنوان** : تماما كما لكل كلمة في الذاكرة لديها عنوان ، كذلك لكل جهاز ، فلذلك يجب على وحدة الإدخال/الإخراج التعرف على العنوان الوحيد لكل طرفية تتحكم بها ، وعلى الجانب الأخر يجب على وحدة الإدخال/الإخراج أن تكون قادرة على أجراء **الإتصالات بالجهاز** .

الإتصالات تتضمن أوامر ، ومعلومات عن الحالة ، وبيانات (الشكل 10.2) . ومن المهام الأساسية لوحدة الإدخال/الإخراج هي **التخزين المؤقت للبيانات** ، والحاجة لهذه الوظيفة أمر واضح حيث أن معدل النقل من/إلى الذاكرة الرئيسية أو **المعالج عالي جداً** ، وهذا المعدل مرتفع على العديد من الأجهزة الطرفية **ويغطي طيفاً واسعاً** من السرعات ، ولذلك ترسل البيانات القادمة من الذاكرة **الرئيسية إلى وحدة الإدخال/الإخراج في معدل سريع ويتم تخزينها مؤقتا في وحدة الإدخال/الإخراج ومن ثم ترسل إلى الجهاز الطرفي** حسب **معدل نقل البيانات الخاصة بالجهاز الطرفي . وفي الاتجاه المعاكس يتم تخزين البيانات مؤقتا حتى لا ترتبط الذاكرة**

في عملية نقل بطيئة ، وبالتالي يجب على وحدة الإدخال/الإخراج أن تكون قادرة على العمل على السرعتين : سرعة الجهاز الطرفي وسرعة الذاكرة ، وبالمثل فإذا كان جهاز الإدخال/الإخراج يعمل بمعدل أعلى من معدل النقل إلى الذاكرة فإن وحدة الإدخال/الإخراج تقوم بعملية التخزين المؤقت التى تحتاج لها .

وأخيراً ، وحدة الإدخال/الإخراج غالبا ما تكون مسئولة عن **إكتشاف الاخطاء** والإبلاغ عن هذه الأخطاء في وقت لاحق للمعالج . وهذه الأخطاء تشمل الأعطال الميكانيكية والكهربائية للجهاز (على سبيل المثال : انحشار الورق ، مسار قرص معطوب) . الاخطاء الأخرى تتضمن التغييرات الغير مقصودة لنمط الخانات المرسلة من الجهاز إلى وحدة الإدخال/الإخراج ، ولذلك إحدى طرق إكتشاف الاخطاء تستخدم للكشف عن أخطاء الإرسال ، ومن أبسط الطرق يمكن إستخدام خانة التماثل لكل حرف/كلمة من البيانات . على سبيل المثال ، رمز الحرف فى المدونة الأبجدية المرجعية الدولية (IRA) يحتل 7 خانات ، ويتم تخصيص الخانة الثامنة كخانة تماثل للسبع خانات الأولى الخاصة بالبيانات . وعندما تستقبل الخانات الثمانية ، وحدة الإدخال/الإخراج تتحقق من التماثل لتحديد ما إذا كان قد حدث خطأ أم لا فى الإرسال وذلك بمقارنة خانة التماثل المنتجة مع المستقبلة .

10.2.2 بنية وحدة الإدخال/الإخراج

وحدات الإدخال/الإخراج تختلف في التعقيد وعدد الأجهزة الخارجية التي يمكن أن تتحكم بها ، والشكل (10.3) يوضح مخطط عام لوحدة الإدخال/الإخراج . الوحدة ترتبط مع باقى الحاسب من خلال مجموعة من خطوط الإشارة (على سبيل المثال: خطوط ناقل النظام) ، والبيانات المنتقلة من وإلى الوحدة يتم تخزينها مؤقتا في مسجل بيانات واحد أو أكثر ، وربما يوجد مسجل حالة أو أكثر يوفر معلومات عن

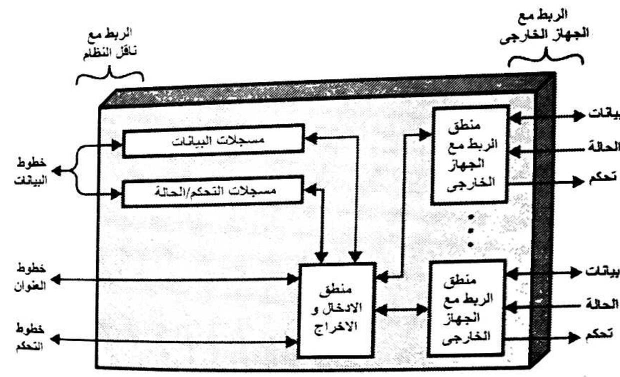
الحالة الحالية للوحدة ، ومسجل الحالة قد يعمل أيضا كمسجل مراقبة وذلك لإستقبال معلومات تفصيلية للتحكم من قبل المعالج . منطق التحكم (المتحكم) داخل وحدة الإدخال/الإخراج يتفاعل مع المعالج من خلال مجموعة من خطوط التحكم حيث يستخدم المعالج خطوط التحكم لاصدار الأوامر إلى وحدة الإدخال/الإخراج . أيضاً، بعض خطوط التحكم يمكن أن تستخدم من قبل وحدة الإدخال/الإخراج (على سبيل المثال : للتحكيم وإشارات الحالة) .

كذلك يجب على وحدة الإدخال/الإخراج أن تكون قادرة على التعرف وإنشاء عناوين للأجهزة التي تتحكم بها ، فكل وحدة إدخال/إخراج لها عنوان وحيد أو مجموعة وحيدة من العناوين (إذا كانت تتحكم فى أكثر من جهاز خارجي) .

وأخيراً ، وحدة الإدخال/الإخراج تحتوي على منطق مادى خاص للربط مع كل جهاز تتحكم به .

وظيفة وحدة الإدخال/الإخراج هى السماح للمعالج بالتعامل مع مجموعة واسعة من الأجهزة الملحقة بطريقة بسيطة ، لذلك هناك تشكيلة واسعة من الخدمات التي يمكن أن تقدمها .

وحدة الإدخال/الإخراج قد تخفي تفاصيل التزامن ، والتنسيق ، والميكانيكية الكهربائية للجهاز الخارجي بحيث أن المعالج يمكن أن يعمل بأوامر قراءة و كتابة بسيطة ، أو ربما بأوامر فتح و غلق ملف . وفي أبسط الصور ، وحدة الإدخال/الإخراج قد تترك الكثير من أعمال التحكم بالجهاز مرئية للمعالج (على سبيل المثال : ترجيع الشريط) .



الشكل (10.3) - المخطط العام لوحدة الإدخال/الإخراج

ويشار عادةً لوحدة الإدخال/الإخراج التى تأخذ على عاتقها معظم أعباء معالجة الإدخال/الإخراج بالتفصيل وتقدم واجهة ربط رفيعة المستوى مع المعالج **كقناة إدخال/اخراج** أو معالج إدخال/إخراج . وحدة الإدخال/الإخراج البدائية جداً والتى تتطلب تحكم تفصيلى تسمى **مُتحكم الإدخال/الإخراج** أو وحدة متحكم الجهاز . وتوجد متحكمات الإدخال/الإخراج عادة في الحواسيب الصغيرة ، وفي حين تستخدم القنوات فى الحاسبات الرئيسية .

في التالى ، سوف نستخدم مصطلح عام وهو وحدة الإدخال/الإخراج بصفة عامة وسوف نستخدم مصطلحات أكثر تحديداً عند الضرورة .

10.3 الإدخال/الإخراج المُبرمج

فى الأساس توجد ثلاث تقنيات لعمليات الإدخال/الإخراج ، أولها الإدخال/الإخراج المُبرمج وفيها يتم تبادل البيانات بين المعالج و وحدة الإدخال/الإخراج ، وذلك بأن

ينفذ المعالج برنامج يُمكنه من التحكم مباشرةً فى عملية الإدخال/الإخراج ، بما في ذلك استشعار حالة الجهاز ، وإرسال اوامر قراءة أو كتابة ، ونقل البيانات . وعندما يصدر المعالج أمر إلى وحدة الإدخال/الإخراج يجب عليه أن ينتظر حتى تنتهى العملية بالكامل . وفى حالة أن المعالج أسرع من وحدة الإدخال/الإخراج يجب عليه الأنتظار وهذا يعتبر إضاعة لوقت المعالج .

التقنية الثانية للإدخال/الاخراج هى بإستخدام المقاطعة . فى الإدخال/الإخراج باستخدام المقاطعة فإن المعالج يصدر أمراً للإدخال/الإخراج ، ثم يستمر فى تنفيذ تعليمات أخرى ، وتتم مقاطعته من قبل وحدة الإدخال/الإخراج عند إكمالها العمل. فى كل من الإدخال/الإخراج المُبرمج والإدخال/الإخراج بإستخدام المقاطعة فإن المعالج مسئول عن إستخراج البيانات من الذاكرة الرئيسية فى حالة الإخراج وتخزين البيانات بالذاكرة الرئيسية فى حالة الإدخال .

البديل هو التقنية الثالثة للإدخال/الاخراج و تعرف بالوصول المباشر للذاكرة (DMA) وفى هذا الوضع ، وحدة الإدخال/الإخراج والذاكرة الرئيسية تتبادلان البيانات مباشرة دون تدخل المعالج .

الجدول (10.1) – تقنيات الإدخال/الإخراج

الأسلوب	بدون مقاطعة	باستخدام المقاطعة
النقل من وحدة الإدخال/الإخراج الى الذاكرة بواسطة المعالج	الإدخال/الإخراج المُبرمج	الإدخال/الإخراج بالمقاطعة
النقل المباشر من وحدة الإدخال/الإخراج الى الذاكرة	---	الوصول المباشر للذاكرة (DMA)

يبين الجدول (10.1) العلاقة بين التقنيات الثلاث الخاصة بالإدخال/الإخراج ، في **هذا القسم** سوف نستكشف الإدخال/الإخراج المُبرمج . وسوف نتطرق **للإدخال/الإخراج** باستخدام المقاطعة والوصول المباشر للذاكرة (DMA) في القسمين التاليين .

10.3.1 نظرة عامة على الإدخال/الإخراج المُبرمج

حينما يكون المعالج مستغرقا في تنفيذ برنامج ما و واجه تعليمة تستلزم **إدخال/إخراج** يقوم بتنفيذ هذه التعليمة عن طريق إصدار أمر إدخال/إخراج إلى **وحدة الإدخال/الإخراج** المناسبة . مع الإدخال/الإخراج المُبرمج فإن وحدة **الإدخال/الإخراج** تقوم بتنفيذ الإجراء المطلوب ثم بعد ذلك تقوم بتعليم خانة مناسبة **في مسجل الحالة** بالوحدة (الشكل – 10.3) ، فوحدة الإدخال/الإخراج لا تقوم بأي **إجراء** آخر لتنبيه المعالج ، بمعنى آخر إنها لا تقاطع عمل المعالج . وهكذا ، فإنه **على عاتق** المعالج أن يتحقق بشكل دوري من حالة وحدة الإدخال/الإخراج حتى **يكتشف** إنتهاء عملية الإدخال/الإخراج .

لشرح تقنية الإدخال/الإخراج المُبرمجة ، سوف ننظر لها أولا من وجهة نظر **الأوامر** الصادرة من المعالج إلى وحدة الإدخال/الإخراج ، ثم من وجهة نظر **تعليمات الإدخال/الإخراج** التي ينفذها المعالج .

10.3.2 أوامر الإدخال/الإخراج

لتنفيذ تعليمة ذات صلة بالإدخال/الإخراج على المعالج أن يُصدر العنوان الذى يحدد وحدة الإدخال/الإخراج والجهاز الخارجي (الطرفية) ، ومن ثم أمر

الإدخال/الإخراج . هناك اربعة أنواع من اوامر الإدخال/الإخراج **التي قد تستقبلها** وحدة الإدخال/الإخراج عند تعاملها مع المعالج :

التحكم : يستخدم لتنشيط الطرفية وإخبارها بما يجب القيام به . **فعلى سبيل المثال ،** وحدة الشريط المغناطيسي قد تأمر بالتراجع أو **التحرك إلى الأمام سجلاً واحداً .** أوامر التحكم مصممة لنوع معين من الاجهزة الطرفية لذلك **فهى تختلف باختلاف** الطرفية .

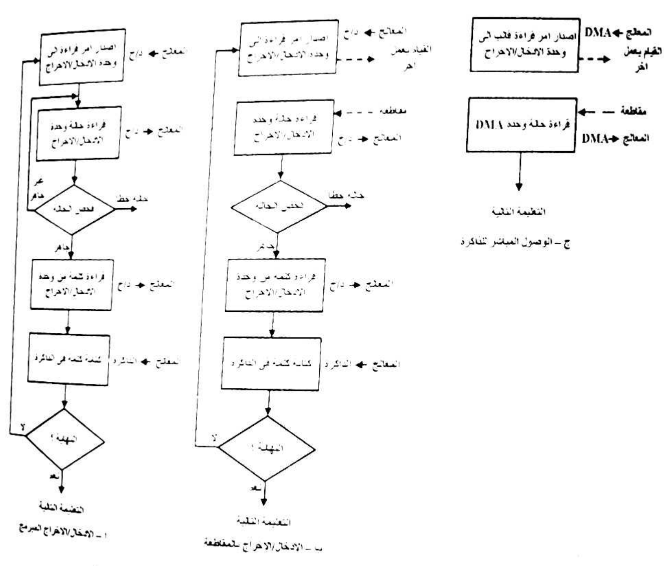
الأختبار : تستخدم لاختبار الظروف المختلفة **(الحالة) لوحدة الإدخال/الإخراج** والأجهزة الطرفية المرتبطة بها . **فالمعالج يريد أن يعرف أن الطرفية المطلوبة** مدعومة من قبل الوحدة ومتاحة للاستخدام ، **وكذلك يريد معرفة ما إذا كانت آخر** عملية إدخال/إخراج قد إنتهت **بنجاح أو حدث بها أخطاء .**

القراءة : تُسبب فى بدء **وحدة الإدخال/الإخراج فى الحصول على البيانات (وحدة** بيانات) **من الطرفية ووضعها في المخزن المؤقت الداخلي (مبين كمسجل بيانات** في الشكل – 10.3) ، **والمعالج يمكنه بعد ذلك الحصول على البيانات من خلال** طلبها **من وحدة الإدخال/الإخراج ومن تم وضعها على متن ناقل البيانات .**

الكتابة : تسبب فى **قيام وحدة الإدخال/الإخراج بأخذ البيانات (خانة أو كلمة) من** ناقل البيانات **وأرسال هذه البيانات في وقت لاحق إلى الطرفية .**

الشكل (10.4 – أ) يعطي مثالاً على إستخدام الإدخال/الإخراج المُبرمج لقراءة قالب من البيانات من جهاز طرفي الى الذاكرة (على سبيل المثال : سجل من شريط مغناطيسى) . فتتم قراءة البيانات التى فى كلمة واحدة (16 خانة) في آن واحد ، وعند قراءة كل كلمة يجب أن يظل المعالج في دورة تدقيق الحالة حتى يتأكد أن الكلمة المطلوبة وصلت وموجودة في مسجل البيانات بوحدة الإدخال/الإخراج .

المخطط الأنسيابي يسلط الضوء على العيب الرئيسي لهذا الأسلوب وهو إنها عملية **إستنزاف** وقت حيث يبقى المعالج مشغول (متابعة) بدون داع .



الشكل (10.4) – التقنيات الثلاث لإدخال/إخراج البيانات

10.3.3 تعليمات الإدخال/الإخراج

مع الإدخال/الإخراج المُبرمج ، هناك تطابق وثيق بين التعليمات ذات الصلة بالإدخال/الإخراج التى يجلبها المعالج من الذاكرة وأوامر الإدخال/الإخراج التى يصدرها المعالج إلى وحدة الإدخال/الإخراج لتنفيذ هذه التعليمات ، بمعنى إنه يمكن مطابقة التعليمات إلى أوامر إدخال/إخراج بسهولة ، وغالبا ما تكون هناك علاقة

مباشرة (واحد الى واحد) وشكل التعليمة يعتمد على الطريقة التي تتم بها **معاملة** الأجهزة الخارجية .

في العادة ، هناك العديد من أجهزة الإدخال/الإخراج متصلة بالنظام من خلال وحدات الادخال/الاخراج ، وكل جهاز يُمنح مُعرف أو عنوان وحيد . وعندما يصدر المعالج امر ادخال/إخراج يحتوي هذا الأمر على عنوان الجهاز المطلوب وبالتالي يجب على كل وحدة إدخال/إخراج مراقبة خطوط العناوين لتحديد ما إذا كان الأمر يخصها ام لا .

عندما يشترك المعالج والذاكرة الرئيسية و وحدة الإدخال/الإخراج فى ناقل عام ، هناك وضعان محتملان للعنونة : الذاكرة المُسقطة أو المنفصلة .

فى الإدخال/الإخراج **بالذاكرة المُسقطة** توجد مساحة عنونة واحدة لمواقع الذاكرة وأجهزة الإدخال/الإخراج . والمعالج يعامل مسجلات الحالة والبيانات لوحدات الإدخال/الإخراج كمواقع فى الذاكرة ويستخدم نفس التعليمات للوصول إلى كل من الذاكرة وأجهزة الإدخال/الإخراج ، لذلك ، على سبيل المثال ، مع 10 خطوط عناوين فإن مجموع مواقع الذاكرة وعناوين الإدخال/الإخراج التى يمكن للمعالج التعامل معها هو $2^{10}=1024$ موقع ، وبأي شكل من أشكال التقاسم أو مجموع المواقع لكل منهما (مجموع عدد مواقع الذاكرة وعدد عناوين أجهزة الإدخال/الإخراج = 1024) . ومع الإدخال/الإخراج بالذاكرة المُسقطة ، هناك حاجة لخطين (خطوط) أحدهما للقراءة وآخر للكتابة على متن الناقل .

فى أسلوب الإدخال/الإخراج **المُنفصل** فإن الناقل به خطوط للقراءة وأخرى للكتابة بالإضافة ذلك خط تحكم بالإدخال والإخراج ، والآن يُحدد خط التحكم ما إذا كان

العنوان يشير إلى موقع ذاكرة أو جهاز إدخال/إخراج ، والمجموع الكامل من العناوين قد يكون متاح لكليهما . مرة أخرى ، مع 10 خطوط عنوان فإن النظام يمكنه دعم 1024 موقع ذاكرة و 1024 عنوان للإدخال/الإخراج وذلك لأن مساحة عناوين الإدخال/الإخراج منفصلة عن الذاكرة .

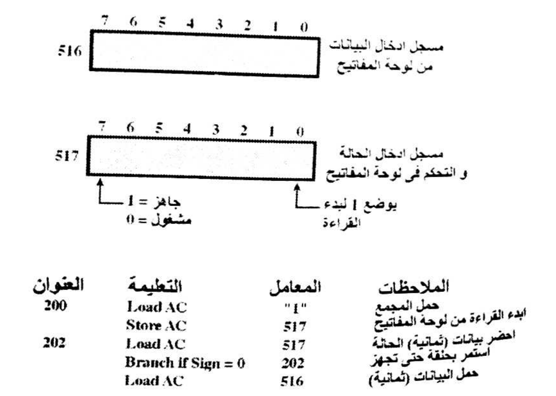
الشكل (10.5) يوضح تقنيتى الإدخال/الإخراج المُبرمج ، فالشكل (10.5 – أ) يبين كيف يبدو ربط جهاز إدخال بسيط مثل طرفية لوحة المفاتيح بإستخدام أسلوب الذاكرة المُسقطة . وبفرض عنوان بطول 10 خانات ، مع ذاكرة بحجم 512 خانة (المواقع 0–511) ، وعناوين إدخال/إخراج تصل الى 512 موقع (المواقع 512–1023) . ويوجد عنوانين مخصصين للمُدخلات من وحدة لوحة المفاتيح ، فالعنوان 516 يشير إلى مسجل البيانات والعنوان 517 يشير إلى مسجل الحالة والذي يعمل أيضا كمسجل تحكم لتلقي أوامر المعالج .

البرنامج الموضح فى الشكل يقراء 8 خانات (ثمان) من البيانات من لوحة المفاتيح إلى مسجل المجمع في المعالج ، ولاحظ أن المعالج يدور فى حلقة حتى تتوفر خانات البيانات .

مع الإدخال/الإخراج المنفصل (الشكل (10.5 – ب)) فإن منافذ الإدخال/الإخراج لا يمكن التواصل معها إلا عن طريق أومر تحكم إدخال/إخراج خاصة ، والتى بدورها تُفعل خطوط تحكم الإدخال/الإخراج على متن الناقل .

بالنسبة لمعظم أنواع المعالجات هناك مجموعة واسعة نسبيا من التعليمات المختلفة التى تؤشر للذاكرة . وبإستخدام الإدخال/الإخراج المنفصل لا يوجد سوى عدد قليل من تعليمات الإدخال/الإخراج ، وبالتالي ميزة الإدخال/الإخراج بالذاكرة المُسقطة

هي أنه يُمكن من إستخدام عدد كبير من التعليمات للإدخال/الإخراج ، مما يسمح ببرمجة أكثر فعالية ، والعيب هو أن مساحة قيمة من عناوين الذاكرة يتم إستخدامها للإدخال/الإخراج ، وكلا من أسلوبى الإدخال/الإخراج بالذاكرة المسقطة والمنفصلة شائعى الأستعمال .



العنوان	التعليمة	المعامل	الملاحظات
200	Load AC	"1"	حمل المجمع
	Store AC	517	ابدء القراءة من لوحة المفاتيح
202	Load AC	517	احضر بيانات (ثمانية) الحالة
	Branch if Sign = 0	202	استمر بحلقة حتى تجهز
	Load AC	516	حمل البيانات (ثمانية)

(أ) الادخال/الاخراج المسقط على الذاكرة

العنوان	التعليمة	المعامل	الملاحظات
200	Load I/O	5	ابدء القراءة من لوحه المفاتيح
201	Test I/O	5	اختبر الانتهاء
	Branch Not Ready	201	استمر بحلقة حتى الانتهاء
	In	5	حمل البيانات (ثمانية)

(ب) الادخال/الاخراج المنفصل

الشكل (10.5) – الإدخال/الإخراج المُسقط والمُنفصل

10.4 الإدخال/الإخراج بالمقاطعة

المشكلة مع الإدخال/الإخراج المُبرمج هو أن المعالج عليه الانتظار وقتا طويلا حتى تجهز وحدة الإدخال/الإخراج المستهدفة لاستقبال أو إرسال البيانات . وأثناء الأنتظار على المعالج أن يستكشف مرارا وتكرارا حالة وحدة الإدخال/الإخراج ، ونتيجة لذلك فإن مستوى أداء النظام بالكامل يتدهور بشدة . بديل ذلك هو أن المعالج يصدر أمر إلى الوحدة ثم ينتقل للقيام ببعض الأعمال المفيدة الأخرى و وحدة الإدخال/الإخراج تقاطع المعالج لطلب الخدمة عندما تكون مستعدة لتبادل البيانات مع المعالج ، ومن تم المعالج ينفذ اجراءات نقل البيانات كما سبق ثم يستأنف العمل السابق .

دعونا الأن ننظر كيف يتم هذا ، أولا من وجهة نظر وحدة الإدخال/الإخراج : في عملية الإدخال فإن وحدة الإدخال/الإخراج تتلقى أمر قراءة من المعالج ، وبناءا على ذلك تقوم بقراءة البيانات من الطرفية التى ترتبط بها ، وبمجرد توفر البيانات في مسجل البيانات بوحدة الإدخال/الإخراج تقوم الوحدة بارسال إشارة مقاطعة الى المعالج عبر خط التحكم ، ومن ثم تنتظر الوحدة حتى يتم طلب البيانات الموجودة بها من قبل المعالج ، وعندما يتم الطلب تقوم الوحدة بوضع البيانات على متن ناقل البيانات وتستعد لعملية الإدخال/الإخراج الأخرى .

من وجهة نظر المعالج فالإدخال كما يلي : المعالج يصدر أمر قراءة ومن ثم ينصرف الى فعل شيء آخر (على سبيل المثال : المعالج قد يعمل على عدة برامج مختلفة في نفس الوقت) ، وفي نهاية كل دورة تعليمة المعالج يتحقق من حدوث مقاطعات (الشكل – 3.12) . وعند حدوث مقاطعة من وحدة الإدخال/الإخراج ، المعالج يحفظ سياق البرنامج الحالى (على سبيل المثال : عدد البرنامج ومسجلات

المعالج) ويعالج المقاطعة . وفي هذه الحالة فإن المعالج يقرأ كلمة من البيانات من وحدة الإدخال/الإخراج ويخزنها في الذاكرة ، ومن ثم يعيد سياق البرنامج الذى كان يعمل (أو البرامج الاخرى) ويستأنف التنفيذ .

الشكل (10.4 – ب) يبين استخدام الإدخال/الإخراج بالمقاطعة لقراءة قالب من البيانات . وقارن هذا مع الشكل (10.4 – أ) . الإدخال/الإخراج بالمقاطعة هو أكثر كفاءة من الإدخال/الإخراج المُبرمج لانه يحد من الانتظار الذى لا داعي له ، ومع ذلك ، الإدخال/الإخراج بالمقاطعة لا يزال يستهلك الكثير من وقت المعالج لأن كل كلمة من البيانات تنقل من الذاكرة الى وحدة الإدخال/الإخراج أو من وحدة الإدخال/الإخراج الى الذاكرة يجب ان تمر عبر المعالج .

10.4.1 معالجة المقاطعة

دعونا ننظر الى دور المعالج في الإدخال/الإخراج بالمقاطعة بمزيد من التفاصيل، فوقوع المقاطعة يتسبب في عدد من الأحداث في كل من عتاد وبرمجيات المعالج . ويبين الشكل (10.6) التسلسل النموذجي لمعالجة المقاطعة ، فعندما يُنهى جهاز عملية المعالجة يكون قد حدث التسلسل التالي من الأحداث المادية :

1. الجهاز يُصدر إشارة مقاطعة إلى المعالج .
2. المعالج يُنهى تنفيذ التعليمة الحالية قبل الاستجابة الى المقاطعة ، وكما هو مبين في الشكل (3.12) .
3. المعالج يختبر المقاطعات ويُحدد ما إذا كانت هناك واحدة أم لا ، ويُرسل إشارة أقرار إلى الجهاز الذي أصدر المقاطعة ، والإقرار يسمح للجهاز بإزالة إشارة المقاطعة .

برمجى

مادى

متحكم الجهاز يصدر المقاطعة

المعالج يكمل تنفيذ التعليمة الحالية

المعالج يصدر اشارة اقرار بالمقاطعة

المعالج يحفظ نسق البرنامج (PSW) و عداد البرنامج فى المكدس

المعالج يحمل قيمة عداد برنامج جديد حسب المقاطعة

حفظ بقية المعلومات عن حالة المعالج

معالجة المقاطعة

استرجاع معلومات عن حالة المعالج

استرجاع قيمة عداد البرنامج و معلومات (PSW) القديمة

الشكل (10.6) – الكيفية المبسطة لمعالجة المقاطعة

4. **المعالج يُحضر الآن لنقل السيطرة إلى روتين المقاطعة . وللبدء فإنه يحتاج إلى حفظ المعلومات اللازمة لأستئناف البرنامج الحالي عند نقطة المقاطعة . الحد الأدنى من المعلومات المطلوبة هو (أ) حالة المعالج والتى توجد فى مسجل يسمى كلمة نسق البرنامج (PSW) ، و (ب) موقع التعليمة التالية للتنفيذ والتى توجد في عداد برنامج وهذه المعلومات تدفع إلى مكدس النظام .**

5. يقوم المعالج الآن بتحميل عداد البرنامج بموقع بداية برنامج مناول/روتين المقاطعة الذى يناسب هذه المقاطعة . وأعتماداً على بنية الحاسب وتصميم نظام التشغيل ، فقد يكون هناك برنامج واحد لكل نوع من المقاطعات ، أو برنامج واحد لكل جهاز ولكل نوع من المقاطعات . إذا كان هناك أكثر من روتين معالجة مقاطعة فيجب على المعالج تحديد أي منها يُنفذ . هذه المعلومات قد تكون في إشارة المقاطعة الأصلية أو قد يصدر المعالج طلباً إلى الجهاز الذي أصدر المقاطعة للحصول على رد يحتوي على المعلومات المطلوبة .

عندما يتم تحميل عداد البرنامج ، المعالج يتقدم الى دورة التعليمة التالية ، والتي تبدأ بجلب التعليمة ، ولأن جلب التعليمة يُحدد بمحتويات عداد البرنامج الذى تم تحميله مسبقاً بموقع بداية برنامج مناول/روتين المقاطعة فالنتيجة هي أنه تم نقل التحكم إلى برنامج روتين المقاطعة ، وتنفيذ هذا البرنامج ينتج العمليات التالية:

6. عند هذه النقطة ، يتم حفظ عداد البرنامج وكلمة نسق البرنامج (PSW) المتعلقة بالبرنامج المتوقف فى مكدس النظام ، ومع ذلك هناك معلومات آخرى تعتبر جزءاً من حالة البرنامج المتوقف ، وعلى وجه الخصوص محتويات مسجلات المعالج يحتاج لحفظها وذلك لأن هذه المسجلات قد يتم إستخدامها من قبل روتين المقاطعة ، ولذلك فكل هذه القيم بالإضافة إلى أي معلومات أخرى عن حالة المعالج تحتاج إلى حفظ . عادةً ، روتين المقاطعة يبدأ بحفظ محتويات كافة المسجلات في المكدس ، ويبين الشكل (10.7 – أ) مثال بسيط على ذلك . ففي هذه الحالة ، تتم مقاطعة برنامج المستخدم بعد التعليمة التى في الموقع (N) ، ويتم دفع محتويات كافة المسجلات بالإضافة إلى عنوان التعليمة التالية (N +1) فى المكدس ، ويتم تحديث مؤشر المكدس ليشير إلى

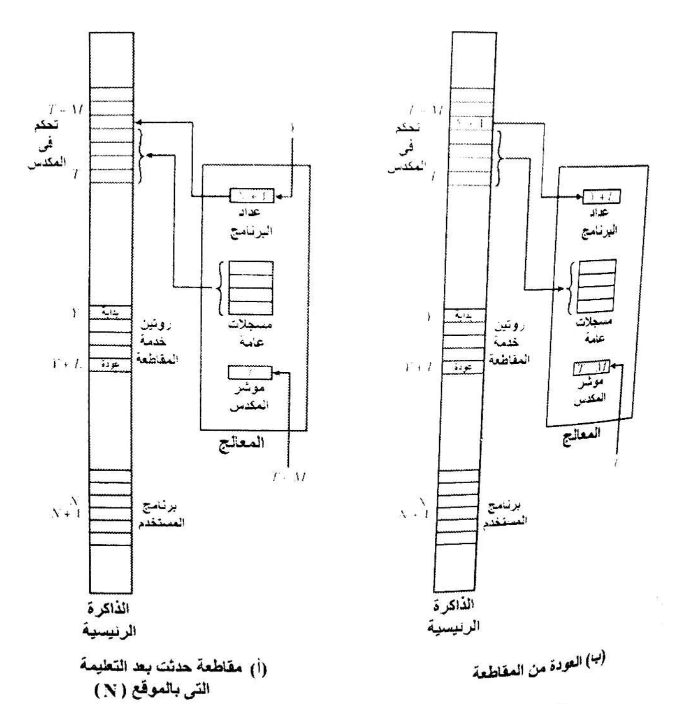
القمة الجديدة للمكدس ، ويتم تحديث عداد برنامج ليشير إلى بداية روتين خدمة **المقاطعة** .

7. **بعد ذلك يقوم روتين خدمة المقاطعة بمعالجة المقاطعة** ، وهذا يشمل أختبار **معلومات الحالة المتعلقة بعملية الإدخال/الإخراج** أو حدث آخر سبب في **المقاطعة ، وقد يتضمن** أيضا إرسال أوامر إضافية أو إقرارات إلى جهاز **الإدخال/الإخراج** (الذى سبب المقاطعة) .

8. **عند انتهاء معالجة المقاطعة** يتم إسترداد قيم المسجلات المحفوظة من المكدس **وإعادة تخزينها** فى مسجلات المعالج (على سبيل المثال : أنظر الشكل 10.7 – ب) .

9. **فى الختام** تتم إستعادة قيم كلمة نسق البرنامج (PSW) وعداد البرنامج من **المكدس ، وينتج** عن ذلك أن التعليمة التالية للتنفيذ ستكون من البرنامج **المتوقف** سابقا .

لاحظ أنه من المهم حفظ كافة المعلومات حول حالة البرنامج المتوقف حتى يُستأنف **في وقت لاحق ، وذلك** لأن المقاطعة ليست روتين/برنامج جزئى يستدعى من **البرنامج ، ولكن المقاطعة** يمكن أن تحدث في أي وقت ، وبالتالي في أي لحظة من **تنفيذ برنامج المستخدم ، وذلك** لان حدوث المقاطعة لا يمكن التنبؤ به .
في الواقع ، قد لا يكون بين البرنامجين (البرنامج المنفذ و روتين خدمة المقاطعة) **أى شيء مشترك وقد ينتميان لتطبيقين أو مستخدمين مختلفين** .



(ب) العودة من المقاطعة

(أ) مقاطعة حدثت بعد التعليمة التى بالموقع (N)

الشكل (10.7) – التغييرات فى الذاكرة والمسجلات عند المقاطعة

10.4.2 معالجة الإدخال/الإخراج المتعدد

تبرز قضيتان فى إنجاز الإدخال/الإخراج بالمقاطعة . أولا ، **لأنه فى الأغلب هناك** عدد متغير من وحدات الإدخال/الإخراج المختلفة ، **فكيف يمكن للمعالج تحديد الجهاز الذي** أصدر المقاطعة ؟ ، **ثانيا ، إذا حدثت مقاطعات متعددة ، فكيف يمكن للمعالج إتخاذ قرار إيها يُنفذ ؟**

لننظر أولا لقضية تحديد الجهاز الذي أصدر المقاطعة ، وهناك أربع أنواع عامة من التقنيات هي الشائعة فى الأستعمال :

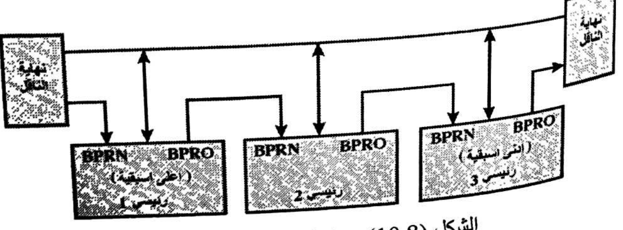
- **خطوط مقاطعة متعددة .**
- **الأنتخاب البرمجى .**
- **سلسلة خاصة (أنتخاب مادى ، مُوجه) .**
- **تحكيم الناقل (مُوجه) .**

النهج الأكثر مباشرة لحل هذه المشكلة هو توفير خطوط مقاطعة متعددة بين المعالج **و وحدات** د/خ (الإدخال/الإخراج) . مع ذلك ، فإنه من غير العملي تخصيص أكثر من بضعة خطوط ناقل أو دبابيس معالج كخطوط مقاطعة ، ونتيجة لذلك حتى إذا ما تم إستخدام خطوط متعددة فمن المحتمل أن كل خط سوف يكون لديه عدة وحدات د/خ مرفقة به ، وهكذا فإن إحدى التقنيات الثلاث الأخرى يجب أن تستخدم لكل خط .

احد البدائل هو **الأنتخاب البرمجى** . فعندما يكتشف المعالج مقاطعة ، فإنه يقفز إلى روتين خدمة مقاطعة مهمته هي إستقصاء كل وحدة د/خ لتحديد أيها تسببت في المقاطعة . وهذا الإستقصاء قد يكون على شكل أمر منفصل (على سبيل المثال : اختبار د/خ) ، وفي هذه الحالة المعالج ينشط خط أختبار د/خ ويضع عنوان معين لوحدة د/خ على خطوط العناوين ، و وحدة د/خ تستجيب إيجابيا إذا هى التى أحدثت المقاطعة . وبدلا من ذلك يمكن أن يكون لكل وحدة د/خ مسجل حالة قابل للعنونة ويقوم المعالج بقراءة مسجل الحالة لكل وحدة د/خ لتحديد الوحدة التى أصدرت المقاطعة ، وعندما يتم تحديد الوحدة الصحيحة يقوم المعالج بالقفز إلى روتين خدمة المقاطعة المحدد بذلك الجهاز .

من سيئات الأنتخاب البرمجى هو أنه مضيعة للوقت . هناك تقنية أكثر كفاءة وهى إستخدام **سلسلة خاصة** ، وفى الواقع يمكن أن تدعى **بالأنتخاب المادى** ، ومثال على تنظيم سلسلة خاصة موضح في الشكل (10.8) .

فى المقاطعات بالأنتخاب المادى فإن جميع وحدات د/خ تشترك فى خط طلب المقاطعة ، وخط إقرار المقاطعة هو مربوط كسلسلة بين وحدات د/خ ، وعندما يتحسس المعالج مقاطعة فإنه يرسل إقرار مقاطعة . وهذه الإشارة تنتشر خلال سلسلة من وحدات د/خ حتى تصل الى الوحدة التى إصدرت طلب المقاطعة و تستجيب الوحدة الطالبة عادةً بوضع كلمة على خط البيانات ، ويشار إلى هذه الكلمة **بالموجه** ، وهى إما أن تكون عنوان وحدة د/خ أو عنوان وحيد آخر ، وفي كلتا الحالتين يستخدم المعالج الموجه كمؤشر إلى روتين خدمة الجهاز المناسب ، وهذا يجنب الحاجة لتنفيذ روتين خدمة مقاطعة عام أولا (لتحديد الجهاز المقاطع من ثم تحديد روتين مقاطعته) وتسمى هذه التقنية **بالمقاطعة المُوجّهة** .



الشكل (10.8) – تحكيم موزع فى تسلسل

هناك أسلوب آخر يستخدم المقاطعات الموجهة ، وهو **تحكيم الناقل** . مع تحكيم الناقل يجب على وحدة د/خ أولا كسب السيطرة على الناقل قبل أن تتمكن من تنشيط

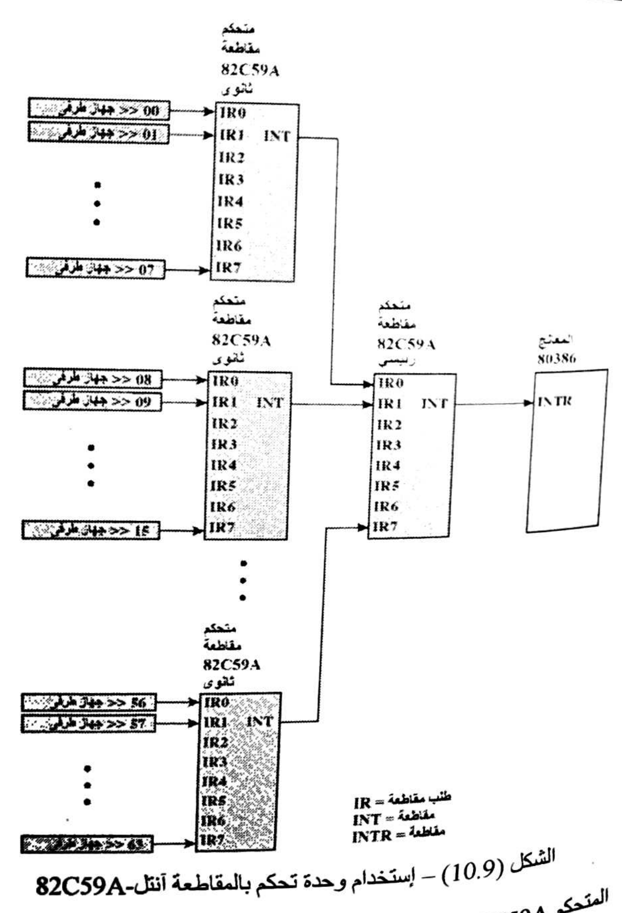
خط طلب المقاطعة ، وهكذا يمكن لوحدة واحدة فقط من تنشيط الخط في وقت واحد. وعندما يكتشف المعالج المقاطعة فإنه يرد على خط اقرار المقاطعة ومن ثم الوحدة الطالبة تضع موجهها على خط البيانات .

التقنيات المذكورة أعلاه هى لتحديد الوحدة الطالبة ، كما أنها توفر وسيلة لتحديد الأسبقيات عندما يكون هناك أكثر من جهاز واحد يطلب خدمة المقاطعة . فى الخطوط المتعددة المعالج يختار خط المقاطعة ذو الأسبقية العليا . مع الانتخاب البرمجي ، الترتيب الذي يتم به إستقصاء وحدات د/خ يحدد أسبقياتها ، وبالمثل فإن ترتيب وحدات د/خ فى السلسلة يحدد أسبقياتها . وأخيراً ، تحكيم الناقل يوظف الأسبقيات وذلك كما نوقش في الفصل الثالث . ونقدم الأن مثالين على بنيات المقاطعة .

10.4.3 وحدة تحكم بالمقاطعة آنتل-82C59A

المعالج آنتل – 80386 يوفرخط طلب مقاطعة واحد (INTR) وخط إقرار بالمقاطعة واحد (INTA) . ولتمكين المعالج 80386 من التعامل مع مجموعة متنوعة من الأجهزة بإسبقيات مختلفة فيتم ربطه مع متحكم مقاطعة خارجي هو 82C59A ، ويتم توصيل الأجهزة الخارجية مع المتحكم 82C59A ، والذي بدوره يرتبط بالمعالج 80386 .

يبين الشكل (10.9) إستخدام 82C59A لربط وحدات د/خ متعددة مع المعالج 80386 . فالمتحكم 82C59A يمكنه التعامل مع ما يصل إلى ثماني وحدات د/خ، وإذا أردنا التحكم فى أكثر من ثماني وحدات نحتاج لترتيب متتالي يُمكن من التعامل مع ما يصل إلى 64 وحدة .



الشكل (10.9) – إستخدام وحدة تحكم بالمقاطعة آنتل-82C59A

المتحكم 82C59A يستعمل فى إدارة المقاطعات حيث أنه يقبل طلبات المقاطعة من الوحدات د/خ المرفقة ، ويحدد المقاطعة الأعلى أسبقية ، ثم ينبه المعالج عن

طريق تنشيط خط (INTR) ، ويُقر المعالج بالمقاطعة عبر خط (INTA) ، وهذا يدفع المتحكم 82C59A إلى وضع معلومات الموجه المناسبة على ناقل البيانات، والمعالج يمكنه بعد ذلك المضي قدماً فى معالجة المقاطعة والتواصل مباشرة مع وحدة د/خ لقراءة أو كتابة البيانات .

المتحكم 82C59A قابل للبرمجة ، والمعالج 80386 يحدد نمط الأسبقيات المستخدم عن طريق وضع كلمة تحكم في 82C59A ، وهذا يُمكن من صيغ المقاطعة التالية :

- متداخلة بالكامل : يتم ترتيب طلبات المقاطعة في أسبقية من 0 (IR0) إلى 7 (IR7) .
- تدوير : في بعض التطبيقات بعض الأجهزة المقاطعة متساوية الاسبقية ، وفي هذا الوضع الجهاز بعد أن يتحصل على الخدمة تتغير أسبقيته وتصبح الأدنى فى المجموعة .
- حجب خاص : يسمح هذا للمعالج بأن يمنع المقاطعات عن أجهزة معينة .

10.4.4 الواجهة الطرفية القابلة للبرمجة أنتل-82C55A

كمثال على وحدة د/خ تُستخدم فى الإدخال/الإخراج المُبرمج والإدخال/الإخراج بالمقاطعة نقدم واجهة إنتل الطرفية القابلة للبرمجة 82C55A. أنتل 82C55A- مبنية علي شريحة واحدة كوحدة د/خ للأغراض العامة مصممة للإستخدام مع معالج إنتل 80386 ، والشكل (10.10) يوضح المخطط العام لها بالإضافة لأسماء 40 مخرج التى تخص مغلف الشريحة .

الجانب الأيمن من الرسم التخطيطي (10.10 – أ) **هو الواجهة الخارجية لأنتل- 82C55A** ، ويبين الخطوط 24 الخاصة بالإدخال/الإخراج **وهى قابلة للبرمجة** من قبل المعالج 80386 بواسطة مسجل تحكم .

المعالج 80386 يمكن ان يُعرف قيم لمسجل التحكم **بحيث يحدد مجموعة متنوعة** من صيغ العمل والترتيبات اللازمة لذلك ، **فالخطوط 24 الخاصة** بالأدخال/الاخراج تقسم إلى ثلاث مجموعات (A, B, C) ، **وكل مجموعة تعمل** بوصفها منفذ د/خ بطول 8 خانات ، بالإضافة إلى ذلك **المجموعة (C) يتم تقسيمها** جزئيا الى مجموعتين (C_A و C_B) لكل منها 4-خانات ، **ويمكن إستخدامها جنباً** الى جنب مع المنفذين (A و B) . والترتيب بهذه الطريقة **يجعل خطوط المجموعة** (C) تحمل إشارات التحكم والحالة .

الشكل (10.10) - الواجهة الطرفية القابلة للبرمجة أنتل-82C55A

...جانب الأيسر من الرسم التخطيطي يبين الربط الداخلي مع ناقل المعالج 80386 ، ...يشمل ناقل بيانات 8- خانات ثنائي الاتجاه (D0 الى D7) وتستخدم لنقل البيانات ...من وإلى منافذ د/خ ونقل معلومات التحكم إلى مسجل التحكم ، والخطين الأثنين ...للعنوان (A0-A1) يحددان أحد منافذ د/خ الثلاثة أو مسجل التحكم ، وعملية نقل البيانات تحدث عندما يتم تنشيط خط أختيار الشريحة (Select Chip) مع أحد خطى القراءة أو الكتابة ، ويستخدم خط التهيئة (Reset) لتهيئة الوحدة .

10.5 الوصول المباشر إلى الذاكرة (DMA)

مع تطور التقنية والحاجة لأساليب تُمكن من تحسين كفاءة نظام الإدخال/الإخراج ومن تم كفاءة نظام الحاسب ككل صممت التقنية الثالثة للإدخال/الإخراج بحيث تتبادل وحدة الإدخال/الإخراج والذاكرة الرئيسية البيانات بدون تدخل المعالج . وسنتطرق في هذا القسم بإيجاز لهذه التقنية .

10.5.1 عيوب الإدخال/الإخراج المُبرمج وبالمقاطعة

رغم أن الإدخال/الإخراج بالمقاطعة أكثر كفاءة من المُبرمج إلا أنه لا يزال يتطلب مشاركة فعالة من المعالج لكى يتم نقل البيانات مابين الذاكرة و وحدة الإدخال/الإخراج ، كذلك أى أنتقال للبيانات يجب أن يمر من خلال المعالج ، لذلك الإدخال/الإخراج المُبرمج وبالمقاطعة يعانى من عيبين متأصليين :

1. معدل نقل الإدخال/الإخراج مقيد بمدى إمكانية المعالج بإختبار وخدمة الجهاز (الطرفية) .
2. المعالج مقيد فى إدارة نقل الإدخال/الإخراج حيث أن العديد من التعليمات يجب تنفيذها لكل عملية نقل إدخال/إخراج (مثال – الشكل 10.5) .

هناك نوع من المفاضلة بين هذين العيبين ، وبأفتراض نقل قالب من البيانات فإن إستخدام الإدخال/الإخراج المُبرمج وتفرغ المعالج لمهمة الإدخال/الإخراج يمكن من نقل البيانات بمعدل عالى نوعا ما ولكن بتكلفة عدم القيام بأى فعل آخر (من قبل المعالج) ، والإدخال/الإخراج بأسلوب المقاطعة يُحرر المعالج لحد ما من تكلفة معدل النقل .

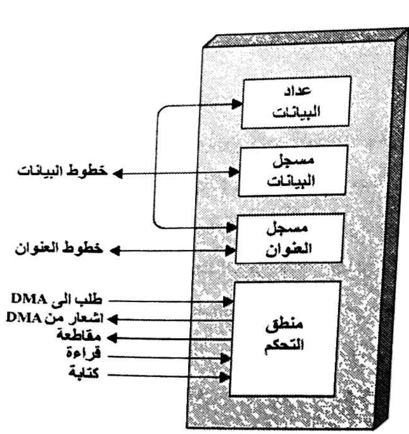
بصفة عامة ، فإن كلتا الطريقتين لهما تأثير سلبي على نشاط المعالج ومعدل النقل على السواء ، ولنقل كميات كبيرة من البيانات مطلوب تقنية أكثر كفاءة وهى **الوصول المباشر للذاكرة (Direct Memory Access)** .

10.5.2 وظيفة الوصول المباشر للذاكرة

الوصول المباشر للذاكرة ينطوي على وجود وحدة إضافية على ناقل النظام وهى وحدة الوصول المباشر للذاكرة (DMA) . فوحدة الوصول المباشر للذاكرة (الشكل 10.11) قادرة على محاكاة المعالج وتولي السيطرة على ناقل النظام بدلاً من المعالج وذلك بغرض القيام بنقل البيانات من وإلى الذاكرة عبر ناقل النظام ، ولهذا السبب يجب أن تستخدم وحدة الوصول المباشر للذاكرة الناقل فقط عندما لا يحتاج إليه المعالج ، أو تجبر المعالج على تعليق العمل مؤقتا ، والأسلوب الأخير هو أكثر شيوعا ويشار إليها بأسم **سرقة دورة** (فترة زمنية) حيث أن وحدة الوصول المباشر للذاكرة في الواقع تسرق دورة (فترة زمنية خاصة) من دورات العمل على الناقل .

عندما يرغب المعالج فى قراءة أو كتابة قالب من البيانات فإنه يصدر أمراً إلى وحدة الوصول المباشر للذاكرة وذلك بإرسال المعلومات التالية إليها :

- **ما إذا كان الطلب للقراءة أو الكتابة** ، وذلك بإستخدام خط تحكم القراءة أو **الكتابة بين المعالج و** وحدة الوصول المباشر للذاكرة .
- **عنوان جهاز الإدخال/الإخراج** المقصود ويُرسل عبر خط البيانات .
- **موقع البداية في الذاكرة** للقراءة منه أو الكتابة فيه ويُرسل عبر خط البيانات **ويحفظ من قبل** وحدة الوصول المباشر للذاكرة فى مسجل العناوين .
- عدد الكلمات التى ستُقرأ أو ستُكتب وتُرسل عبر خط البيانات وتحفظ فى مسجل عداد البيانات .

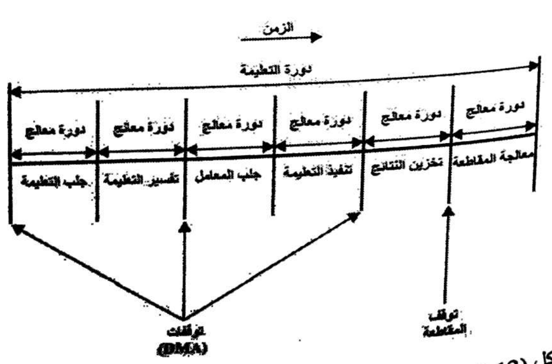


الشكل (10.11) – مخطط نموذجى لوحدة الوصول المباشر للذاكرة

ثم يستمر المعالج في تنفيذ أعمال اخرى ، ومع تفويض وحدة الوصول المباشر للذاكرة لهذه العملية من الإدخال/الإخراج تقوم وحدة الوصول المباشر للذاكرة بنقل هذا القالب من البيانات بالكلمة ، ومباشرة من أو الى الذاكرة دون المرور بالمعالج.

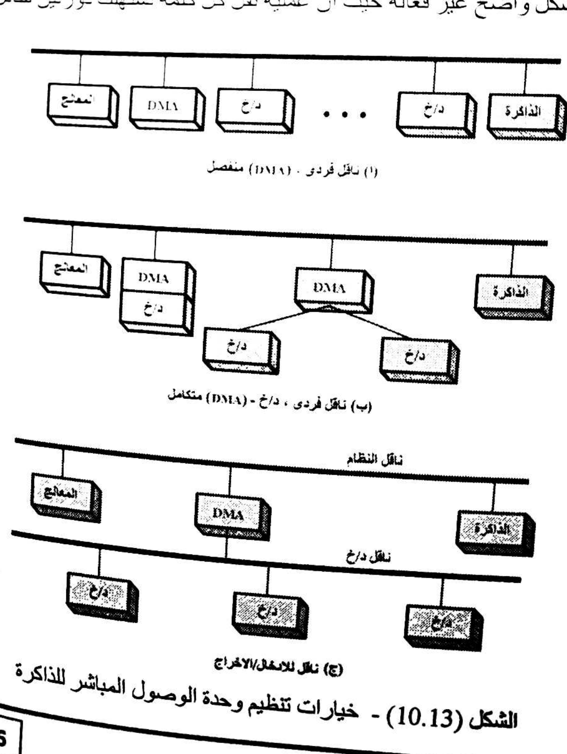
وعند إنتهاء عملية النقل تُرسل وحدة الوصول المباشر **للذاكرة إشارة مقاطعة** للمعالج وبذلك المعالج يشارك فى بداية ونهاية عملية النقل فقط (**أنظر للشكل 10.4** – ج) .

الشكل (10.12) يُظهر أين يجوز تعليق عمل المعالج فى **دورة التعليمة ، ففي كل** حالة يتم تعليق المعالج قبل الحاجة إلى إستخدام الناقل ، **ثم تقوم وحدة الوصول** المباشر للذاكرة بنقل كلمة واحدة وتُرجع التحكم إلى المعالج . **ولاحظ أن هذا** الاجراء ليس مقاطعة حيث المعالج يحفظ السياق ويفعل شيئا آخر . **بتعليق المعالج** مؤقتاً لدورة ناقل واحدة هذا يسبب أن المعالج يعمل ببطء أكثر ، **ومع ذلك ففى حالة** نقل كلمات متعددة وحدة الوصول المباشر للذاكرة **هى أكثر كفاءة بكثير من** الإدخال/الإخراج المُبرمج أو بالمقاطعة .



الشكل (10.12) – توقفات المقاطعة و وحدة الوصول المباشر للذاكرة أثناء دورة التعليمة

آلية تنظيم عمل وحدة الوصول المباشر للذاكرة يمكن ان يتم بطرق متنوعة والشكل **(10.13) يبين بعض** هذه التنظيمات . في المثال الاول ، جميع الوحدات تشترك **في نفس ناقل النظام** و وحدة الوصول المباشر للذاكرة تعمل كمعالج بديل ، ويستخدم **الإدخال/الإخراج** المُبرمج لتبادل البيانات بين الذاكرة و وحدة الإدخال/الإخراج من **خلال وحدة الوصول** المباشر للذاكرة ، هذه التهيئة رغم أنها قد تكون غير مكلفة **ولكنها بشكل واضح غير** فعالة حيث أن عملية نقل كل كلمة تستهلك دورتين للناقل.



(أ) ناقل فردي ، (DMA) منفصل

(ب) ناقل فردي ، د/خ - (DMA) متكامل

(ج) ناقل للإدخال/الإخراج

الشكل (10.13) - خيارات تنظيم وحدة الوصول المباشر للذاكرة

يمكن خفض عدد دورات الناقل المطلوبة **بشكل كبير من خلال دمج وحدة الوصول** المباشر للذاكرة مع وظائف الإدخال/الإخراج ، **وكما يشير الشكل (10.13 – ب)،** وهذا يعني ان هناك مسارا بين وحدة الوصول المباشر للذاكرة و **وحدة أو أكثر من** وحدات الإدخال/الإخراج لا يتضمن ناقل النظام ، **وربما عتاد وحدة الوصول** المباشر للذاكرة قد يكون في الواقع جزءا من وحدة الإدخال/الإخراج ، **أو قد تكون** وحدة منفصلة تتحكم في وحدة أو أكثر من وحدات الإدخال/الإخراج .

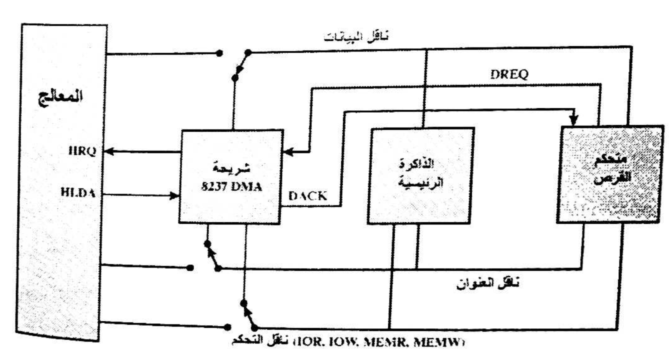
هذا المفهوم يمكن أن يطور من خلال ربط وحدات الإدخال/الإخراج **إلى وحدة** الوصول المباشر للذاكرة بإستخدام ناقل خاص بالإدخال/الإخراج **(الشكل 10.13** – ج) ، وهذا يقلل من عدد واجهات الإدخال/الإخراج في **وحدة الوصول المباشر** للذاكرة الى واحدة فقط ، وهذا يوفر إمكانية التوسع بسهولة . **في كل من هذه الحالات** (الشكل 10.13 – ب ، ج) ناقل النظام المشترك مابين **وحدة الوصول المباشر** للذاكرة والمعالج والذاكرة يستخدم من قبل **وحدة الوصول المباشر للذاكرة لتبادل** البيانات مع الذاكرة فقط ، وتبادل البيانات بين **وحدة الوصول المباشر للذاكرة و** وحدات الإدخال/الإخراج يتم بدون إستعمال ناقل النظام .

10.5.3 متحكم الوصول المباشر للذاكرة (DMA) - آنتل 8237A

متحكم الوصول المباشر للذاكرة (DMA) آنتل 8237A **يعمل كواجهة لعائلة** المعالجات 80x86 لتقديم إمكانية الوصول المباشر للذاكرة **الرئيسية من نوع** الذاكرة التفاعلية (DRAM) ، والشكل (10.14) **يوضح موقع وحدة الوصول** المباشر للذاكرة . فعندما تحتاج وحدة الوصول المباشر للذاكرة **لأستخدام ناقل** النظام (البيانات، العنوان، التحكم) لنقل البيانات ، **فإنها ترسل إشارة (HOLD)** إلى المعالج ، ويستجيب المعالج بإشارة (HLDA) – **(إقرار HOLD) مقبول**

إلى أن وحدة الوصول المباشر للذاكرة يمكنها إستخدام الناقل . على سبيل المثال ، إذا كانت وحدة الوصول المباشر للذاكرة ستنقل قالب بيانات من الذاكرة إلى القرص، فإنها ستقوم بما يلي :

1. الجهاز الطرفي (مثل مُتحكم القرص) سوف يطلب خدمة الوصول المباشر للذاكرة عن طريق رفع اشارة (DREQ) – (طلب DMA) عالياً (منطق مرتفع) .



الشكل (10.14) - إستخدام وحدة الوصول المباشر للذاكرة (8237) لناقل النظام

2. **وحدة الوصول المباشر للذاكرة سترفع إشارتها (HRQ) عالياً (طلب) لكى تشعر وحدة المعالجة المركزية من خلال دبوس (HOLD) أنها تحتاج إلى إستخدام ناقل النظام .**

3. وحدة المعالجة المركزية سوف تنهي دورة الناقل (ليس بالضرورة للتعليمة الحالية) وتستجيب لطلب الوصول المباشر للذاكرة برفع إشارة (HDLA) عالياً (اقرار HOLD) ، وبهذا تُخبر وحدة الوصول المباشر للذاكرة 8237 أنه يُمكنها المضي قدما وإستخدام ناقل النظام لأداء مهمتها . وإشارة (HOLD) يجب أن تظل عالية التفعيل طالما وحدة الوصول المباشر للذاكرة تقوم بمهمتها .
4. وحدة الوصول المباشر للذاكرة تُفعل إشارة (DACK) – (إقرار DMA) والذي سيخبر الجهاز الطرفي إنها ستبدأ في نقل البيانات .
5. وحدة الوصول المباشر للذاكرة تبدأ بنقل البيانات من الذاكرة إلى الطرفية بوضع عنوان أول ثمان بيانات من القالب على ناقل العناوين وتُفعل اشارة (MEMR) ، وبالتالي تُقرأ ثمان من الذاكرة إلى ناقل البيانات ؛ بعد ذلك تُفعل (IOW) للكتابة إلى الطرفية ، بعد ذلك تُنقص وحدة الوصول المباشر للذاكرة العداد وتزيد مؤشر العنوان وتكرر هذه العملية حتى يصل العداد للصفر و تنتهى المهمة .
6. بعد إنتهاء وحدة الوصول المباشر للذاكرة من وظيفتها سوف تُعطل (HRQ) مما ينبه وحدة المعالجة المركزية بإمكانية إستعادة التحكم بناقل النظام .

بينما تستخدم وحدة الوصول المباشر للذاكرة ناقل النظام لنقل البيانات **يبقى المعالج** خاملاً ، بالمثل فعندما يستخدم المعالج الناقل **وحدة الوصول المباشر للذاكرة تبقى** خاملة . وحدة الوصول المباشر للذاكرة **8237 تعرف بأنها متحكم (الوصول المباشر للذاكرة) السريع ، وهذا يعني أن البيانات التي يتم نقلها من موقع لأخر لا** تمر عبر شريحة الوصول المباشر للذاكرة **ولا تخزن في شريحة الوصول المباشر**

للذاكرة وبالتالى وحدة الوصول المباشر للذاكرة يمكنها فقط نقل البيانات بين منفذ وموقع بالذاكرة وليس بين منفذى إدخال/إخراج أو موقعي ذاكرة . ومع ذلك يمكن لشريحة الوصول المباشر للذاكرة من إجراء نقل من ذاكرة إلى ذاكرة عبر مسجل.

الشريحة (8237) تحتوي على أربع قنوات للوصول المباشر للذاكرة يمكن برمجتها بشكل مستقل ، ويمكن تفعيل أي منها في أي لحظة ، وهذه القنوات مرقمة 0 ، 1 ، 2 ، و 3 . الشريحة (8237) لديها مجموعة من خمسة مسجلات تحكم أومر للبرمجة والتحكم بتشغيل قنوات الوصول المباشر للذاكرة .

10.6 قنوات ومعالجات الإدخال/الإخراج

مع تطور نظام الحاسب و ظهور تقنيات جديدة تطورت آليات الإدخال/الإخرج بشكل أصبحت معه أكثر أستقلالية وذات كفاءة أعلى وهذا حسن من أداء النظام . في هذا القسم نستعرض الآلية المعاصرة للإدخال/الإخرج وهى قنوات و معالجات الإدخال/الإخراج .

10.6.1 تطور وظائف الإدخال/الإخراج

كلما تطورت أنظمة الحاسب زادت مكوناته الفردية تطوراً و تعقيداً ، ويتجلى هذا بوضوح فى وظائف الإدخال/الإخراج التى شهدت بالفعل جزءاً من هذا التطور ، ومراحل هذا التطور يمكن تلخيصها في مايلي :

1. **المعالج يتحكم مباشرة فى الجهاز الطرفي ، ويلاحظ هذا فى الأجهزة البسيطة المتحكم بها بواسطة المعالج .**
2. **إضافة وحدة إدخال/إخراج أو مُتحكم ، وهنا يستخدم المعالج الإدخال/الإخراج المُبرمج بدون مقاطعة ، وبهذه الخطوة المعالج بعيد إلى حد ما من تفاصيل الأرتباط مع جهاز خارجي محدد .**

3. إستخدام نفس التهيئة المذكورة فى الخطوة (2) **ولكن يتم الآن توظيف** المقاطعات ، ولا حاجة للمعالج أن **يضيع وقتاً في إنتظار عملية** الإدخال/الإخراج لأنجازها وبالتالي تزداد الكفاءة .
4. تمنح وحدة الإدخال/الإخراج إمكانية الوصول المباشر إلى الذاكرة **عبر وحدة** الوصول المباشر للذاكرة (DMA) ويمكن الآن نقل قالب من البيانات من أو إلى الذاكرة دون إشراك المعالج إلا في بداية ونهاية النقل .
5. تعزيز وحدة الإدخال/الإخراج لتصبح معالجاً في حد ذاتها وبمجموعة تعليمات خاصة مصممة للإدخال/الإخراج . **وحدة المعالجة المركزية توجه معالج** الإدخال/الإخراج لتنفيذ برنامج إدخال/إخراج محفوظ في الذاكرة ، **ومعالج** الإدخال/الإخراج **يجلب وينفذ هذه التعليمات دون تدخل وحدة المعالجة** المركزية ، **وهذا يسمح لوحدة المعالجة المركزية لتحديد سلسلة من أنشطة** الإدخال/الإخراج وأن تُقاطع فقط عندما يتم تنفيذ هذه السلسلة بالكامل .
6. وحدة الإدخال/الإخراج لديها ذاكرة محلية بنفسها ، **وهي في الواقع حاسب** بحد ذاتها . **بهذه المعمارية مجموعة كبيرة من أجهزة الإدخال/الإخراج يمكن** التحكم بها مع الحد الأدنى من تدخل وحدة المعالجة المركزية . **الإستخدام** الشائع لمثل هذه المعمارية **هو التحكم فى التواصل مع المحطات التفاعلية ،** ومعالج الإدخال/الإخراج **يهتم بمعظم المهام التي تدخل في التحكم فى** المحطات .

على طول هذا المسار التطوري فإن المزيد والمزيد من وظائف الإدخال/الإخراج يتم تنفيذها دون تدخل وحدة المعالجة المركزية ، و وحدة المعالجة المركزية تتخلص من المزيد من أعباء الإدخال/الإخراج مما يحسن من الأداء . ففى العقدين

ـاضيتين (6-5) حدث تغير كبير مع إدخال مفهوم وحدة إدخال/إخراج قادرة على ـفيذ برنامج . ففى الفقرة 5 ، غالبا ما يشار إلى وحدة الإدخال/الإخراج بأنها **قناة ـخال/إخراج** . وفى الفقرة 6 ، غالبا ما يستخدم مصطلح معالج إدخال/إخراج . ـعموما ، فى بعض الأحيان المصطلحين يستعملان فى كلتا الحالتين ، وفي ما يلي ـسوف نستخدم مصطلح **قناة إدخال/إخراج** .

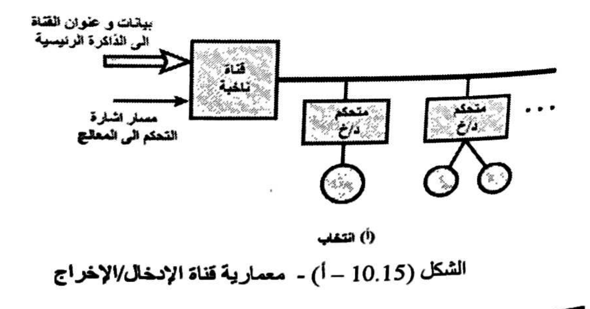
10.6.2 خصائص قنوات الإدخال/الإخراج

قناة الإدخال/الإخراج تمثل أمتداداً لمفهوم وحدة الوصول المباشر للذاكرة ، فقناة الإدخال/الإخراج لديها القدرة على تنفيذ تعليمات الإدخال/الإخراج وذلك يمنحها إمكانية التحكم الكامل فى عمليات الإدخال/الإخراج . في نظم الحاسب التى بها مثل هذه الوحدة فإن وحدة المعالجة المركزية لا تنفذ التعليمات ، بل يتم تخزين هذه التعليمات في الذاكرة الرئيسية ليتم تنفيذها من قبل معالج خاص (خاص بالإدخال/الإخراج) موجود في قناة الإدخال/الإخراج ، وهكذا فإن وحدة المعالجة المركزية تبدء عملية النقل بإعطاء تعليمات لقناة الإدخال/الإخراج بتنفيذ برنامج في الذاكرة . وهذا البرنامج سوف يحدد الجهاز أو الأجهزة ، ومنطقة أو مناطق من الذاكرة للتخزين ، والأولويات ، والإجراءات التي يتعين إتخاذها فى حالات أخطاء معينة ، وقناة الإدخال/الإخراج تتبع هذه التعليمات والضوابط فى نقل البيانات .

هناك نوعين شائعين من قنوات الإدخال/الإخراج ، وكما هو موضح في الشكل (10.15) . **القناة الناخبة** تتحكم فى أجهزة متعددة عالية السرعة وهى – في وقت ما – تختص فى عملية نقل البيانات لأحدى تلك الأجهزة فقط ، وهكذا فإن قناة الإدخال/الإخراج تنتخب جهاز واحداً لعملية نقل البيانات . ويتم التعامل مع كل

جهاز ، أو مجموعة صغيرة من الأجهزة بواسطة متحكم أو وحدة إدخال/إخراج ، وهى مثل وحدات الإدخال/الإخراج التى تم مناقشتها سابقا ، وبهذا فإن قناة الإدخال/الإخراج تعمل بدلا عن وحدة المعالجة المركزية في التحكم فى هذه المتحكمات .

القناة المشتركة يمكنها التعامل مع أجهزة متعددة فى نفس الوقت في عملية الإدخال/الإخراج ، ففى الأجهزة المنخفضة السرعة الأشتراك فى ثُمانَ يُمكن من أستقبال أو أرسال حرف في زمن سريع لأجهزة متعددة . على سبيل المثال ، دفق من الحروف الناتج من ثلاثة أجهزة بمعدلات عمل مختلفة (حرف لكل جهاز) وبدفق فردي لكل واحد منها مثل ، . . . . $A_1A_2A_3A_4$ ، . . . . $B_1B_2B_3B_4$ ، و . . . . $C_1C_2C_3C_4$ يمكن أن يعامل مثل $A_1B_1C_1A_2C_2A_3B_2C_3A_4$ ، وهكذا. للأجهزة العالية السرعة الأشتراك فى قالب (مجموعة من الثمانات مع بعض) يسمح بتداخل قوالب بيانات من أجهزة متعددة .



الشكل (10.15 – أ) - معمارية قناة الإدخال/الإخراج

(ب) مشاركة

الشكل (10.15 – ب) - معمارية قناة الإدخال/الإخراج

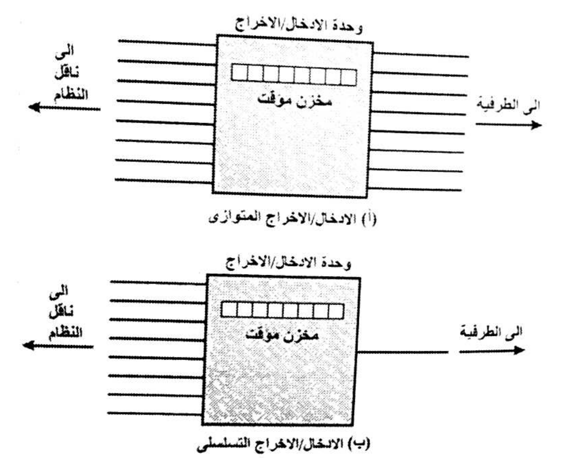
10.7 الروابط الخارجية

في هذا القسم سنتطرق بإيجاز للروابط الخارجية مابين وحدات الإدخال/الإخراج و المحيط الخارجي (الطرفيات/الأجهزة الخارجية) .

10.7.1 أنواع الروابط

ربط الطرفية مع وحدة الإدخال/الإخراج يجب أن يكون متلائما مع طبيعة وعمل الطرفية ، فأحد الخصائص الرئيسية للرابط هى ما إذا كان تسلسلياً أو متوازياً (الشكل – 10.16) . ففى الرابط المتوازي هناك عدة خطوط تربط وحدة الإدخال/الإخراج مع الطرفية ، ويتم نقل عدة خانات فى وقت واحد كما لو أن خانات كلمة واحدة يتم نقلها عبر ناقل البيانات . وفي الرابط التسلسلي هناك خط

واحد فقط يستخدم فى نقل البيانات ، وفى هذه الحالة يجب أن تنقل الخانات بالتسلسل بحيث تنقل فى كل مرة خانة واحدة فقط . يستخدم الرابط المتوازي فى العادة للأجهزة الطرفية العالية السرعة ، مثل الشريط والقرص ، في حين أن الرابط التسلسلي تقليدياً يستخدم للطابعات والمنافذ . ومع الجيل الجديد من الروابط التسلسلية العالية السرعة فإن الروابط المتوازية أصبحت أقل شيوعاً .



الشكل (10.16) - الإدخال/الإخراج التسلسلى و المتوازى

وفى الحالتين (التسلسلى و المتوازى) ، يجب على وحدة الإدخال/الإخراج الدخول في حوار مع الطرفية .

بصورة عامة ، الحوار لعملية كتابة هو كما يلي :

1. **تُرسل وحدة الإدخال/الإخراج إشارة تحكم طالبة الإذن لإرسال البيانات .**

2. تُقر الطرفية الطلب .
3. وحدة الإدخال/الإخراج تنقل البيانات (كلمة واحدة او قالب أعتمادا على الطرفية) .
4. الطرفية تُقر بإستلام البيانات .

عملية القراءة تتم على نحو مماثل .

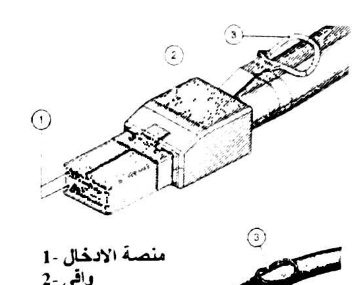
أساس العمل فى وحدة الإدخال/الإخراج هو أن المخزن الموقت الداخلي في الوحدة يُمكنه تخزين البيانات التى يتم تمريرها مابين الطرفية وبقية النظام . ويسمح المخزن المؤقت لوحدة الإدخال/الإخراج بتعويض عن الاختلافات في السرعة مابين ناقل النظام والخطوط الخارجية .

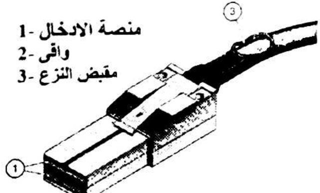
10.7.2 الرابط الفردي والمتعدد

الربط مابين وحدة الإدخال/الإخراج بنظام الحاسب والأجهزة الخارجية يمكن أن يكون إما من فردي (نقطة – إلى – نقطة) أو متعدد (متعدد النقاط) . فالرابط الفردي (نقطة – إلى – نقطة) يوفر خط مخصص مابين وحدة الإدخال/الإخراج والجهاز الخارجي . نموذجياً ، الوصلات الفردية فى الأنظمة الصغيرة (الحواسيب الشخصية ومحطات العمل) تشمل تلك الرابطة للوحة المفاتيح ، وللطابعة ، وللمودم الخارجي، والمثال نموذجي لمثل هذا الرابط هو مواصفات الرابط (EIA-232).

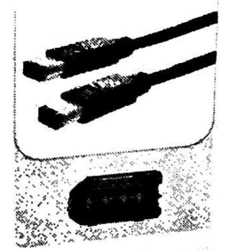
أزدادت أهمية الروابط الخارجية المتعددة النقاط نتيجة إستخدمها فى دعم أجهزة التخزين الخارجية الكبيرة (القرص والشريط) وأجهزة الوسائط المتعددة (الأقراص المدمجة والفيديو والسمعية) ، وهذه الروابط المتعددة النقاط هي فى الوقع ناقلات خارجية بنفس منطق الناقلات التى ناقشنها فى الفصل (3) . وأمثلة على هذه

الروابط فى أنظمة الحاسب المعاصرة هى (FireWire) و (InfiniBand) والشكل الخارجي لمقابسها موضح فى الشكل (10.17) والشكل (10.18) .

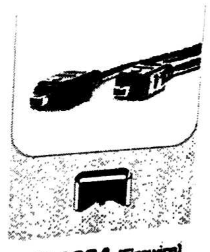




الشكل (10.17) - وصلة (InfiniBand)



IEEE 1394 (Firewire)
6 مسامير



IEEE 1394 (Firewire)
4 مسامير

الشكل (10.18) - وصلة (FireWire)

مصطلحات مهمة

سرقة دورة	Cycle Stealing
الوصول المباشر للذاكرة	Direct Memory Access (DMA)
تقنية ربط سلكي تسلسلي	Firewire
تقنية ربط سلكي عالي السرعة	Infiniband
مقاطعة	Interrupt
الادخال/الإخرج بالمقاطعة	Interrupt-Driven I/O
قناة ادخال/إخراج	I/O Channel
أمر ادخال/إخراج	I/O Command
وحدة ادخال/إخراج	I/O Module
معالج إدخال/اخراج	I/O Processor
إدخال/إخراج تسلسلي	Serial I/O
إدخال/إخراج مسقط على الذاكرة	Memory-Mapped I/O
قناة مشتركة	Multiplexor Channel
إدخال/إخراج متوازى	Parallel I/O
جهاز ملحق/طرفى	Peripheral Device
إدخال/إخراج مبرمج	Programmed I/O
إدخال/إخراج منفصل/معزول	Isolated I/O
ناخب قناة	Selector Channel
جهاز عرض فيديو	VDTs
ثُمانَ – 8 خانات ثنائية	Byte
الأبجدية المرجعية الدولية	International Reference Alphabet (IRA)
كلمة حالة البرنامج	Program Status Word (PSW)
تقنية إتصال تسلسلى لنقل بيانات	(Electronics Industries Association) EIA-232/RS-232
الرابط الخارجي	External Interface

أسئلة للمراجعة

1. أذكر التصنيفات الرئيسية الثلاث للأجهزة الطرفية ؟.
2. ماهى المهام الأساسية لوحدة الإدخال/الإخراج والشكل النموذجى لها ؟.
3. وضح الإدخال/الإخراج بأسلوب المقاطعة ؟.
4. وضح عمل وشكل وحدة الوصول المباشر للذاكرة ؟.
5. ماهى التقنيات الثلاث الاساسية للإدخال/الإخراج بالحاسب؟.
6. عندما حدوث مقاطعة من جهاز خارجى ، كيف يحدد المعالج اي جهاز أصدر المقاطعة ؟.
7. أثناء سيطرة وحدة الوصول المباشر على الناقل مالذى يقوم به المعالج؟.
8. ماهى مميزات الإدخال/الإخراج المسقط على الذاكرة مقارنة بالإدخال/الإخراج المنفصل ؟.
9. المعالج أنتل 8088 يستخدم نسقين لتعليمة الإدخال/الإخراج ، فالأول رمز العملية المكون من 8 خانات ثنائية يعرف عملية الإدخال/الإخراج ثم الحقل الذى يليه من 8 خانات به عنوان المنفذ المستعمل للعملية . وفى النسق الثانى رمز العملية يشير الى أن عنوان المنفذ موجود في المسجل (DX) وهو بعرض 16 خانة . فكم عدد المنافذ التي يمكن للمعالج 8088 التعامل معها في كل نسق ؟ .
10. المعالج (Z8000) عنده قدرة الإدخال/الإخراج بنقل قالب وهى تحت إشراف المعالج المباشر . تعليمة نقل القالب تُعرف مسجل عنوان المنفذ (Rp)، وعداد البرنامج (Rc) و مسجل الوجهة (Rd) . مسجل الوجهة يحتوى على عنوان الموقع في الذاكرة الرئيسية الذى سيخزن به أول ثمان تمت قراءته من منفذ الدخول ، والمسجل (Rc) عبار عن مسجل عام بطول 16 خانة . فكم هو حجم القالب الذى يمكن نقله ؟.
11. معالج يفحص حالة خرج جهاز إدخال/إخراج كل 20 ملى ثانية ، وواجهة ربط الجهاز تتضمن منفذين : أحدهما للحالة و الأخر لخرج البيانات . فكم الزمن المستغرق من المعالج لتحقق وخدمة الجهاز مع علم أن معدل

نبضة المعالج هو 8 ميغا هيرتز ؟. أفترض أن دورة التعليمة تأخذ 12 دورة نبضية.

12. أفترض أن نظام حاسب يستخدم طريقة الإدخال/الإخراج بالمقاطعة مع جهاز معين ينقل في البيانات بمتوسط 8 كيلو ثمان/ثانية و بوتيرة مستمرة.

أ- مع أفتراض أن معالجة روتين المقاطعة تستغرق 100 ميكروثانية (الزمن من لحظة القفز لروتين معالجة المقاطعة ، ثم تنفيذه ، ومن ثم العودة الى البرنامج الرئيسي) . حدد كم هو الجزء من وقت المعالج المستغرق من قبل هذا الجهاز إذا كان يقاطع لكل ثمان .

ب- والآن أفترض أن الجهاز له مخزنين مؤقتين كل منهما بسعة 16 ثمان ويقاطع المعالج كلما أمتلاء أحداهما . طبعا في هذه الحالة زمن معالجة المقاطعة أطول عنه سابقاً نتيجة أن روتين المقاطعة يجب أن ينقل 16 ثمان (سعة المخزن المؤقت) . أثناء تنفيذ روتين المقاطعة فإن زمن نقل كل ثمان يأخذ 8 ميكرو ثانية من المعالج . حدد كم هو الجزء من وقت المعالج المستغرق من قبل هذا الجهاز في هذه الحالة .

ت- أستمراراً لماسبق ، أفترض الآن أن المعالج له تعليمة إدخال/إخراج نقل قالب مثل ألتي موجودة بالمعالج (Z8000) ، وهذه تسمح لروتين المقاطعة بنقل كل ثمان من قالب في 2 ميكرو ثانية . حدد كم هو الجزء من وقت المعالج المستغرق من قبل هذا الجهاز في هذه الحالة .

13. وحدة وصول مباشر للذاكرة (DMA) تنقل أحرف (حرف = 8 خانات ثنائية) إلى الذاكرة الرئيسية بإستخدام أسلوب سرقة الدروة ، علماً بأن الجهاز الخارجي يرسل البيانات بمعدل 9600 خانة ثنائية/ثانية والمعالج يجلب التعليمات بمعدل 1 مليون تعليمة لكل ثانية (1 MIPS) . فكم هو زمن تَبَاطُؤ (تَأَخُّر) المعالج نتيجة هذا النشاط لوحدة الوصول المباشر للذاكرة ؟.

المصادر والمراجع

المصادر والمراجع

1. Computer Organization and Architecture, 8th , Williams Stallings , Pearson Prentice Hall.
2. Essentials of Computer Organization and Architecture, 2nd , Linda Null & Julia Lobur, Jones & Bartlett Learning.
3. Computer Organization and Design: the Hardware/Sofwrae interface, David A. Patterson & John L. Hennessy, 4th, The Morgan Kaufmann.
4. http://mcs.uwsuper.edu/sb/101/Module6/cpusim.html, Date: 1/5/2016, Time:17.00 .
5. http://www.mta.ca/~amiller/cs1611/labs/lab7/lab7_applet.html, Date: 1/5/2016, Time:17.00 .
6. http://www.science.smith.edu/~jcardell/Courses/CSC103/CPUsim/cpusim.html, Date: 1/5/2016, Time:17.00 .
7. **http://www.cs.gordon.edu/courses/cps111/Notes/Machine%20Language/cpu-sim/cpusim.html, Date: 1/5/2016, Time:17.00 .**